

METHODS AND SYSTEMS FOR DATA OBSCURATION

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority to U.S. Provisional Application No. 63/544,735, titled "ANONYMITY IMPLEMENTATION SYSTEM" and filed on October 18, 2023, and to U.S. Provisional Application No. 63/566,752, titled "SYSTEM FOR DYNAMIC DATA ROUTING AND PRIVATIZING" and filed on March 18, 2024, each of which is incorporated by reference herein in its entirety.

TECHNICAL FIELD

[0002] Various embodiments concern computer programs and associated computer-implemented techniques for data security.

BACKGROUND

[0003] In today's digital landscape, users interact with numerous online platforms and applications, each requiring personal data, communication information, or sensitive credentials. Exposing this information poses security risks, especially after data breaches. Existing solutions lack a unified approach to dynamically protecting user data.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] Figure 1 illustrates a network environment that includes a data obscuration platform that is executed by a computing device.

[0005] Figure 2 illustrates an example of a computing device that is able to obscure data associated with users.

[0006] Figures 3A, 3B, and 3C illustrate examples of communication routing between users.

[0007] Figures 4A and 4B illustrate examples of communication routing

between users and platforms.

[0008] Figure 5 illustrates an example of communication routing using communication contents.

[0009] Figures 6A, 6B, 6C, 6D, 6E, and 6F illustrate examples of a data obscuration platform.

[0010] Figures 7A, 7B, 7C, 7D, and 7E illustrate examples of data obscuration on a third-party platform.

[0011] Figure 8 illustrates a process for obscuring credentials associated with users.

[0012] Figure 9 includes an illustration of inputs and outputs of a model.

[0013] Figure 10 includes a flow diagram of a process for obscuring communication identifiers associated with users.

[0014] Figure 11 includes a flow diagram of a process for obscuring credentials associated with users.

[0015] Figure 12 is a block diagram illustrating an example of a processing system that can perform at least some operations described herein.

[0016] Various features of the technology will become more apparent to those skilled in the art from a study of the Detailed Description in conjunction with the drawings. In the drawings, embodiments are illustrated by way of example and not limitation for the purpose of illustration. Those skilled in the art will recognize that alternative embodiments may be employed without departing from the principles of the present disclosure. Accordingly, while specific embodiments are shown in the drawings, the technology is amenable to various modifications.

DETAILED DESCRIPTION

[0017] User privacy and security are increasingly critical as identity theft, data breaches, and fraud have increased in prevalence. When users' personal and sensitive information are compromised by malicious actors, the consequences are severe, with users' sensitive data and identities being particularly vulnerable. Malicious actors can exploit breached and on-sale data to commit fraud, drain accounts, or make unauthorized purchases, leading to significant harm to individuals. Additionally, when personal identity information is stolen or impermissibly purchased, it can be used to open new accounts, commit theft, or allow cyber-stalking, leaving users with long-lasting damage. Protecting users' data is therefore not just a technical issue, but a critical safeguard for users' safety in an increasingly digital world.

[0018] User privacy protections have relied on tools like encryption, firewalls, and secure passwords to ensure that data is secure. Encryption ensures that data transmitted over networks remains unreadable to anyone without the proper decryption key, while firewalls act as barriers between internal networks and external threats. Password protocols protect user accounts by forming a layer of access control. While these tools may increase the security of certain platforms with which a user interacts, they do not provide a user with a comprehensive strategy for protecting personal data across all platforms, websites, or payment methods. By relying on platforms' security measures, users are left vulnerable in the event of a data breach caused by a third party, an employee error, or an intentional breach by a bad actor.

[0019] Introduced here is an approach for enhancing user privacy and digital communications by protecting personal data across platforms with which users interact. The system may have a database of multiple phone numbers owned by the system, which may be assigned by multiple phone carriers. The system may then assign these numbers to individual users. In some embodiments, the system may assign each number for use in specific scenarios by multiple users, with the unique ability to reutilize each number across businesses or individuals. Each user may elect to retain such user's personal phone number, but the system may

dynamically route or assign an owned phone number (or multiple) to the user's number. As such, the system effectively privatizes or otherwise protects against exposure of the user's personal, underlying phone number from external entities, thereby safeguarding personally identifiable data from unauthorized access or misuse.

[0020] As the system can dynamically route communications, when a user inputs a user's personal phone number into a service provider's system, the system may automatically delete this entry and replace it with an owned phone number (or automatically enter an owned phone number in a service provider's system), thereby allowing the system to store information so as to forward all calls/messages directed to the owned phone number to the user's own phone number. In some embodiments, the system may allow for multi-user overlays/layering. The phone number owned by the system may be used simultaneously by multiple users for different service providers. In some embodiments, the system intelligently identifies and routes incoming messages or calls to the appropriate user based on the originating service provider. For example, the system may have the capacity to note which user entered information into a specific service provider and may route all information from that specific service provider to the correct user's personal phone number. In some embodiments, the deliberate overlap of assigned identifiers introduces ambiguity, complicating external profiling efforts.

[0021] In addition to receiving calls and messages, users may initiate communications from the assigned phone numbers. The system may discern the intended recipient of such outbound communications, ensuring that the user's actual phone number remains undisclosed during the process. This can allow for multiple individuals in a single group to appear to be sending messages from a single owned phone number, with all personal numbers being obscured in the process. Moreover, the system offers various customizable settings to enhance user experience and privacy. Users may opt to block calls during specific times, such as weekends, or set up lists to prevent communications from certain service

providers or only permit calls from certain numbers. This flexibility allows users to tailor their communication preferences according to individual needs.

[0022] In some embodiments, users have the option to select preferred modes of communication, such as text-only or call-only options. This feature may be particularly useful for users who wish to limit the type of communications they receive from service providers, thereby enhancing their control over their personal data. Additionally, the system may incorporate features to protect users from potential fraud. Users may report suspicious activities, such as fraudulent calls or messages, directly through the system. This proactive approach to security helps to track incoming communications and sources of communications to determine if such sources are fraudulent or suspected of being fraudulent.

[0023] For the purpose of illustration, embodiments may be described with reference to phone numbers. However, the principles of dynamic data routing and concealment extend beyond phone numbers, as the system can adapt to protect other forms of personal data, such as email addresses, credit card information, and other types of data. This versatility makes the system applicable in various domains in which data privacy may be a concern. For example, the system may automatically update user credentials across various websites, databases, or merchant systems with a single click or step.

[0024] In some embodiments, the system employs a process that conceals or masks users' existing credentials, such as usernames, passwords, email addresses, phone numbers, or payment cards, substituting them with altered credentials. In particular, the system may identify a user profile, on a third-party platform, that has the user's actual credentials stored thereon. The system may retrieve instructions for obscuring the user's credentials. For example, the instructions may be platform-specific such that the system is able to navigate a third-party platform to update the user's credentials. In some embodiments, the system may generate the instructions using a generative AI algorithm, for example,

when the system encounters a new platform for which no existing instructions exist.

[0025] The instructions may guide the system through a login process for the user using the actual credentials associated with the user. To do so, the system may encounter various authentication and verification prompts that must be satisfied in order to complete the login. The system can, in various embodiments, retrieve authentication codes from email accounts, text messages, or authentication applications associated with the user or stored on the user's devices. Once logged in, the system may replace the user's actual credentials with different credentials. For example, the system may retrieve different credentials that are obscured or otherwise secure and may update the user's profile to include the different credentials. The system may remove the actual credentials from the profile such that they are not accessible to third parties.

[0026] Deployment of the system is envisioned on platforms regardless of whether users' credentials are already established. Where user credentials are already established, the system may necessitate specific permissions or additional access from users to modify their credentials. As this introduces complexity and resistance in the user experience, the system can still operate efficiently, ensuring that the obscuration of credentials is executed with effectiveness and precision. Credentials may thus be safeguarded against risks such as fraud, identity theft, privacy violations, and data breaches, as users' existing credentials are not stored by the websites, databases, or merchant systems.

[0027] A user interface will be a part of the system, providing real-time updates on statuses, credentials, or statuses of websites, databases, or merchant systems the user has obscured credentials associated therewith. The interface, which may be embedded within a web application, may allow users the flexibility to customize and control the actions performed by the browser extension (engine). It may ensure that users are kept informed and engaged, updating them on the success

or any potential issues encountered during the obscuration process and prompting necessary user actions when required.

[0028] Moreover, embodiments may be described in the context of computer-executable instructions for the purpose of illustration. However, aspects of the approach may be implemented via hardware or firmware instead of, or in addition to, software. As an example, the system may be embodied as a computer program that offers a user interface, allowing easy access and management of settings. Users may configure preferences, view communication logs, and access security features. This ease of use ensures that the system is accessible to a broad range of users. Further, the system may include data analytics capabilities, allowing users to track and analyze their communication patterns. This feature may provide insights into usage trends, security threats, and other relevant metrics.

Terminology

[0029] References in this description to “an embodiment” or “one embodiment” means that the feature, function, structure, or characteristic being described is included in at least one embodiment of the technology. Occurrences of such phrases do not necessarily refer to the same embodiment, nor are they necessarily referring to alternative embodiments that are mutually exclusive of one another.

[0030] Unless the context clearly requires otherwise, the terms “comprise,” “comprising,” and “comprised of” are to be construed in an inclusive sense rather than an exclusive or exhaustive sense (i.e., in the sense of “including but not limited to”). The term “based on” is also to be construed in an inclusive sense rather than an exclusive or exhaustive sense. Thus, unless otherwise noted, the term “based on” is intended to mean “based at least in part on.”

[0031] The terms “connected,” “coupled,” and variants thereof are intended to include any connection or coupling between two or more elements, either direct or indirect. The connection/coupling can be physical, logical, or a combination thereof. For example, objects may be electrically or communicatively coupled to one another despite not sharing a physical connection.

[0032] The term “module” may refer to software components, firmware components, or hardware components. Modules are typically functional components that generate one or more outputs based on one or more inputs. As an example, a computer program may include multiple modules responsible for completing different tasks or a single module responsible for completing all tasks.

[0033] When used in reference to a list of multiple items, the term “or” is intended to cover all of the following interpretations: any of the items in the list, all of the items in the list, and any combination of items in the list.

Overview of Data Obscuration Platform

[0034] Figure 1 illustrates a network environment that includes a data obscuration platform 102 that is executed by a computing device 104. An individual (also referred to as a “user”) can interact with the data obscuration platform 102 via interfaces 106. For example, a user may be able to access an interface through which to monitor platforms and services for which the user’s data has been obscured. As another example, a user may be able to view alerts that such user’s data has been compromised and may be able to take actions in response. Some interfaces 106 may be designed to allow for the review of information acquired, derived, or produced by the data obscuration platform 102, while other interfaces 106 may be designed to facilitate engagement between users and platforms (e.g., to facilitate users obscuring data on various platforms).

[0035] As shown in Figure 1, the data obscuration platform 102 can reside in a network environment 100. Thus, the computing device 104 on which the data obscuration platform 102 resides can be connected to one or more networks 108A-B. Depending on its nature, the computing device 104 may be connected to a personal area network (“PAN”), local area network (“LAN”), wide area network (“WAN”), metropolitan area network (“MAN”), or cellular network. For example, if the computing device 104 is a computer server, then the computing device 104 may be accessible to users via respective mobile phones that are connected to the Internet via LANs. Data to be examined by the data obscuration platform 102 may

be generated or acquired by respective mobile phones, computing devices, or other devices. In some embodiments, the computing device 104 may be associated with, and accessible to, a user – in which case the computing device 104 may be connected to a server system 110 that is responsible for supporting the data obscuration platform 102. In such embodiments, the computing device 104 may be a mobile phone, tablet computer, or other computing device.

[0036] The data obscuration platform 102 may incorporate various architectures. For example, the data obscuration platform 102 may take the form of a browser extension, which can act as the client-side interface, interacting directly with websites in the user's browser, capturing real-time data, and facilitating user interactions. In some embodiments, the data obscuration platform 102 may be a desktop application, which may provide additional computational resources and system-level integrations, enhancing performance for resource-intensive tasks, managing background processes, and securing storage of user data. In some embodiments, the data obscuration platform 102 may take the form of a backend service, which may host generative AI models and processing algorithms, store user preferences, encrypted credentials, and action histories, and manage communication between the browser extension, desktop app, and AI modules.

[0037] Additionally or alternatively, the computing device 104 may be connected to one or more other computing devices over a short-range wireless connectivity technology, such as Bluetooth®, Near Field Communication ("NFC"), Wi-Fi® Direct (also referred to as "Wi-Fi P2P"), and the like. As an example, the data obscuration platform 102 may be embodied as a mobile application that is executed by a mobile phone. In such embodiments, the mobile phone may be communicatively connected – via a wireless communication channel – to a source from which to acquire data. The data may alternatively be obtained from another computer program executing on the mobile phone. For example, the data may be acquired from another mobile application executing on the mobile phone or the operating system of the mobile phone.

[0038] The interfaces 106 may be accessible via a web browser, desktop application, mobile application, or another form of computer program. For example, a user may be able to access interfaces through which information regarding such user's data or obscured data can be provided, viewed, or altered via a web browser, desktop application, mobile application, or another form of computer program. Through these interfaces, the data obscuration platform 102 may present feedback for improving the user's security by taking additional steps to obscure data.

[0039] The data obscuration platform 102 may be executed by a cloud computing service. Thus, the computing device 104 may be representative of a computer server that is part of a server system 110. Often, the server system 110 is comprised of multiple computer servers. These computer servers can include different types of data (e.g., sets of instructions for obscuring data, information about platforms, user data, etc.), algorithms for processing incoming data, models (e.g., generative artificial intelligence (AI) models, natural language processing (NLP) models, large language models (LLMs), etc.). Those skilled in the art will recognize that this data may also be distributed among the server system 110 and one or more computing devices. For example, some data that is input by, or related to, users may be stored on, and processed by, their own computing devices for security or privacy purposes. As a specific example, data regarding a user's actual (i.e., not obscured) credentials, payment information, communication identifiers, or other data may be stored locally while the obscured data may be transmitted (e.g., to the server system 110).

[0040] Components of the data obscuration platform 102 may also be hosted locally. That is, part of the data obscuration platform 102 may reside on the computing device used to access one of the interfaces 106. For example, the data obscuration platform 102 may be embodied as a mobile application executing on a mobile phone as mentioned above. Note, however, that the mobile application may be communicatively connected to the server system 110 on which other components of the data obscuration platform 102 are hosted.

[0041] Figure 2 illustrates an example of a computing device 200 that is able to obscure data associated with users. As shown in Figure 2, the computing device 200 can include a processor 202, memory 204, display mechanism 206, and communication module 208. Each of these components is discussed in greater detail below.

[0042] Those skilled in the art will recognize that different combinations of these components may be present depending on the nature of the computing device 200. For example, if the computing device 200 is a computer server that is part of a server system (e.g., server system 110 of Figure 1), then the computing device 200 may not include the display mechanism 206. Conversely, if the computing device 200 is a mobile phone, then the computing device 200 can include the display mechanism 206.

[0043] The processor 202 can have generic characteristics similar to general-purpose processors, or the processor 202 may be an application-specific integrated circuit ("ASIC") that provides control functions to the computing device 200. As shown in Figure 2, the processor 202 can be coupled to all components of the computing device 200, either directly or indirectly, for communication purposes.

[0044] The memory 204 can be comprised of any suitable type of storage medium, such as static random-access memory ("SRAM"), dynamic random-access memory ("DRAM"), electrically erasable programmable read-only memory ("EEPROM"), flash memory, or registers. In addition to storing instructions that can be executed by the processor 202, the memory 204 can also store data generated by the processor 202 (e.g., when executing the modules of the data obscuration platform 212). Note that the memory 204 is merely an abstract representation of a storage environment. The memory 204 may be comprised of actual integrated circuits (also called "chips").

[0045] The display mechanism 206 can be any mechanism that is operable to visually convey information to a user. For example, the display mechanism 206

can be a panel that includes light-emitting diodes (“LEDs”), organic LEDs, liquid crystal elements, or electrophoretic elements. As further discussed below, outputs produced by the data obscuration platform 212 (e.g., through execution of its modules) can be posted to the display mechanism 206 for review by a user of the computing device 200.

[0046] The communication module 208 may be responsible for managing communications external to the computing device 200. The communication module 208 can be wireless communication circuitry that is able to establish wireless communication channels with other computing devices. Examples of wireless communication circuitry include 2.4 gigahertz (“GHz”) and 5 GHz chipsets compatible with Institute of Electrical and Electronics Engineers (“IEEE”) 802.11 – also referred to as “Wi-Fi chipsets.” Alternatively, the communication module 208 may be representative of a chipset configured for Bluetooth, NFC, and the like. Some computing devices – like mobile phones, tablet computers, and the like – are able to wirelessly communicate via separate channels, while other computing devices – like watches and fitness trackers – tend to wirelessly communicate via a single channel. Accordingly, the communication module 208 may be one of multiple communication modules implemented in the computing device 200, or the communication module 208 may be the only communication module implemented in the computing device 200.

[0047] The nature, number, and type of communication channels established by the computing device 200 – and more specifically, the communication module 208 – can depend on (i) the sources from which data is received by the data obscuration platform 212 and (ii) the destinations to which data is transmitted by the data obscuration platform 212. Assume, for example, that the data obscuration platform 212 resides on a server. In such embodiments, the communication module 208 can communicate with devices 210A-N external to the computing device 200 from which to obtain data. This data may include user data, platform information, sets of instructions, or any combination thereof. Examples of user data may include passwords, usernames, email addresses, phone numbers, home

addresses, work addresses, security questions and answers, payment information, or other data associated with the user. Examples of platform information may include domain names, metadata, images, or other information relating to a third-party platform.

[0048] For convenience, the data obscuration platform 212 is referred to as a computer program that resides within the memory 204. However, the data obscuration platform 212 may be comprised of software, firmware, or hardware that is implemented in, or accessible to, the computing device 200. In accordance with embodiments described herein, the data obscuration platform 212 can include a processing module 214, training module 216, obscuration module 218, routing module 220, and graphical user interface (“GUI”) module 222. These modules may be integral parts of the data obscuration platform 212, or these modules may be logically separate from the data obscuration platform 212 but operate “alongside” it. These modules enable the data obscuration platform 212 to obscure and protect user data.

[0049] The processing module 214 can process data that is obtained by the data obscuration platform 212 into a format that is suitable for the other modules. For example, the processing module 214 can apply operations to data acquired from the devices 210A-N in preparation for analysis by the other modules of the data obscuration platform 212. For example, the processing module 214 can filter or alter the data, such that the data can be more readily analyzed. As another example, the processing module 214 may parse different types of data (e.g., user data and platform data) or data provided via different mechanisms (e.g., data directly uploaded to data obscuration platform 212 and data obtained by data obscuration platform 212 from memory 204 or another computer program executing on the computing device 200) in order to temporally arrange this data, thereby ensuring that any insights gleaned through analysis of this data are temporally sensible. Such an approach may be helpful in ensuring that changes – for example, credentials entered into a user profile – are properly accounted for.

[0050] In some embodiments, the data obscuration platform 212 can apply a model suite to information associated with a user in order to generate a set of instructions for updating a user's credentials in connection with a third-party platform. The model suite can include generative AI models, NLP models, LLMs, or other models. The training module 216 may be responsible for training these models. The training module 216 may train, for example, a machine learning (ML) model to generate sets of instructions for navigating a website using supervised learning. In supervised learning, the training data may include platform website information or images as input and a desired output, such as a set of instructions for navigating the website to update a user profile. A representation of the third-party platform website can be provided to the model. Output from the model can be compared to the desired output for that website and, based on the comparison, the model can be modified, such as by changing weights between nodes of the neural network or parameters of the functions used at each node in the neural network (e.g., applying a loss function). After modifying the model in this manner, the model can be trained to evaluate new platforms. In some embodiments, the training module 216 may use other methods of training modules, as discussed in greater detail below.

[0051] The obscuration module 218 may function to obscure user data by implementing a variety of techniques designed to protect user privacy and enhance security. One function of the obscuration module 218 may be the replacement of user credentials on various platforms with different credentials. When a user logs into a service, the module may intercept the login request and substitute the user's actual credentials with a set of system-generated credentials. These credentials may be mapped back to the user's real credentials within the data obscuration platform 212. This approach may prevent the user's actual login information from being stored or exposed on external platforms, significantly reducing the risk of credential theft or misuse.

[0052] Additionally, the obscuration module 218 may employ data masking techniques to obscure sensitive information within data streams. For example,

when transmitting user data to third-party services, the module may replace sensitive fields such as Social Security numbers (SSNs), credit card details, or personal addresses with masked values. These masked values may be reversible only within the secure confines of the data obscuration platform 212, ensuring that even if intercepted, the data may be meaningless to unauthorized parties. By using any of these methods or a combination of methods, the obscuration module 218 may provide a robust layer of security, safeguarding user data from various threats and ensuring privacy in digital interactions.

[0053] The routing module 220 may route communications to a user via a different communication identifier. For instance, instead of using the user's actual email address or phone number, the module may generate and use a different email address or phone number. This different communication identifier may then be mapped to the user's real contact information within the data obscuration platform 212 using various techniques, ensuring that external parties only interact with the different communication identifier, thereby protecting the user's true identity and contact details from exposure.

[0054] The routing module 220 may dynamically generate these different communication identifiers, such as secure phone numbers or email addresses, each time a new connection between the user and an entity is formed. For example, the routing module 220 may assign different identifiers to a user in connection with various relationships between the user and other entities. For instance, a user may begin contact with another user, which may prompt the data obscuration platform 212 (e.g., obscuration module 218) to generate a different phone number to facilitate communications between these two users. As another example, the user may begin contact with a third-party platform, which may prompt the obscuration module 218 to generate a different email address to facilitate communications between the user and the third-party platform.

[0055] In some embodiments, when a third party attempts to contact the user via their primary phone number, the routing module 220 may intercept the call or

message and redirect it to a secure phone number. This secure number may then forward the communication to the user's actual phone number. The user may receive the call or message as if it were sent directly to their primary number, but the third party may only be aware of the secure number.

[0056] Moreover, the routing module 220 may integrate with other security features to provide comprehensive protection. For example, it may work in conjunction with the obscuration module 218 to replace user credentials and mask sensitive data within communications. By combining these functionalities, the system may offer a robust solution for safeguarding user information, ensuring that both the content and the routing of communications are protected from unauthorized access and potential threats. This multi-layered approach may be particularly beneficial in scenarios where users need to interact with multiple external parties while maintaining a high level of privacy and security.

Approaches to Obscuring User Data Through "Cloaking"

[0057] Introduced here is an approach for enhancing user privacy and digital communications by protecting personal data across platforms with which users interact. The system may have a database of multiple phone numbers owned by the system, which may be assigned by multiple phone carriers. The system may then assign these numbers to individual users. In some embodiments, the system may assign each number for use in specific scenarios by multiple users, with the unique ability to reutilize each number across businesses or individuals. Each user may elect to retain their personal phone number, but the system may dynamically route or assign an owned phone number (or multiple) to the user's number. As such, the system effectively privatizes or otherwise protects against exposure of the user's personal, underlying phone number from external entities, thereby safeguarding personally identifiable data from unauthorized access or misuse. Because the system assigns one or more owned phone numbers to the user's phone number, thereby disguising the user's phone number, the system may be described as "cloaking" the user – and more specifically, the user's phone number

from other individuals or businesses.

[0058] In some embodiments, two users may wish to communicate without revealing their personal identifiers, such as phone numbers, to each other. This scenario presents a significant challenge, as traditional communication methods inherently expose personal identifiers, which can lead to privacy breaches, unwanted contact, or misuse of information. In some embodiments, the data obscuration platform may enable multiple users to communicate without revealing the users' actual identifiers. For example, the data obscuration platform may route communications through different, intermediary identifiers, such as assigned phone numbers. This intermediary identifier can act as a proxy, masking the users' real phone numbers while allowing them to interact seamlessly. When one user initiates contact, the system intercepts the communication and reroutes it through the assigned identifier to the intended recipient. This process ensures that neither user's actual phone number is exposed at any point. The intermediary identifier can be dynamically generated and managed by the system, providing a secure means of communication.

[0059] As an example, the data obscuration platform may receive a first request to facilitate communication between a first user and a second user. In some embodiments, the data obscuration platform may receive a first input that is representative of a first request to facilitate communication. In some embodiments, the first and second users may belong to a plurality of users of the network-accessible platform (e.g., the data obscuration platform). The first user may be associated with a first identifier and the second user may be associated with a second identifier.

[0060] Identifiers may include government-issued identification numbers such as SSNs, passport numbers, and driver's license numbers. Financial identifiers, such as bank account numbers, credit card numbers, and tax identification numbers, are used for financial transactions and tax reporting. Biometric identifiers, including fingerprints, facial recognition data, and retinal scans, provide

unique physical characteristics for secure authentication and access control. Additionally, digital identifiers like IP addresses, device IDs, and MAC addresses are used to track and manage user interactions in online environments. Usernames and account IDs on various platforms also serve as identifiers, linking individuals to their digital profiles and activities.

[0061] In some embodiments, identifiers may include communication identifiers. Communication identifiers of users may encompass a variety of personal information elements that facilitate contact and interaction. These identifiers may include a user's phone number, which can be used for voice calls, text messages, and two-factor authentication processes. Another communication identifier is the email address, serving as a primary means for electronic communication, account recovery, and receiving notifications. Additionally, communication identifiers may extend to social media handles or usernames on platforms like Twitter, LinkedIn, or Facebook, enabling direct messaging and social interactions. Instant messaging IDs, such as those used on WhatsApp, Telegram, or Skype, also fall under this category, providing real-time communication channels. In some cases, physical mailing addresses may be considered communication identifiers, particularly for sending official documents or marketing materials.

[0062] In some embodiments, the request received by the data obscuration platform may indicate that one or more of the users desires to obscure their identifier. The request to obscure a user's identifier may be communicated to the data obscuration platform through various channels and formats. One method is through a secure dashboard (e.g., as shown in Figures 6A-6F), where users can log in and submit their requests via a dedicated form, specifying the identifiers the users wish to obscure, such as email addresses, phone numbers, or SSNs. In some embodiments, users may send an encrypted email to a support team, detailing the request and providing necessary authentication information. In some cases, the request may be initiated through an API call from an integrated application, where the user's preferences are automatically transmitted to the data

obscuration platform. Additionally, users may use a mobile app associated with the data obscuration service, through which they can navigate to a privacy settings section and select the identifiers they want to obscure. For organizations managing multiple users, batch requests can be submitted via secure file uploads or automated scripts, listing all identifiers to be obscured for a group of users.

[0063] The data obscuration platform may retrieve a third identifier from a database that includes identifiers that are dynamically assignable to, and reclaimable from, the plurality of users. For example, these identifiers can be assigned to users for specific purposes, such as verifying accounts, facilitating anonymous communication, or conducting secure transactions. In some embodiments, the identifiers can be assigned to users in relation to specific relationships (e.g., with other users, with third-party platforms, etc.). In some embodiments, each identifier can be assigned to multiple users in connection with separate relationships. For example, the same identifier can be used between a first user and a first entity (e.g., a first platform) and, simultaneously, between a second user and a second entity (e.g., a second platform). In some embodiments, identifiers may be reclaimed and reassigned to other users, ensuring efficient utilization of resources. In some embodiments, a single identifier may be reused in different scenarios by the same user at the same time.

[0064] By using randomized phone numbers, the system enhances privacy and security, as these numbers do not reveal any personal information about the user and can be easily rotated or deactivated to prevent misuse. Additionally, the ability to reclaim and reassign identifiers helps maintain a clean and organized database, reducing the risk of identifier exhaustion and ensuring that the system can scale to accommodate a growing number of users. This approach is particularly valuable in environments where user anonymity and data protection are paramount, such as in online marketplaces, social media platforms, and secure communication services.

[0065] The data obscuration platform may assign the third identifier to the first

and second users such that communications between the first and second identifiers are routed via the third identifier. In particular, communications that originate from the first identifier and are directed to the third identifier are routed to the second identifier, without revealing the second identifier to the first user. Similarly, communications that originate from the second identifier and are directed to the third identifier are routed to the first identifier, without revealing the first identifier to the second user.

[0066] As an illustrative example, a first user and a second user may be assigned a new identifier (e.g., a new phone number) to facilitate communications between them. The new phone number may be different from either phone number associated with either user. The data obscuration platform may receive, from the first user, a second request to direct a communication via the third identifier to the second user. In some embodiments, the data obscuration platform may receive second input that is representative of the second request to direct a communication via the third identifier to the second user. For example, the first user may attempt to call the second user using the new phone number shared between them. The data obscuration platform may then direct the communication to the second identifier of the second user via the third identifier without revealing the first identifier of the first user to the second user and without revealing the second identifier of the second user to the first user. For example, the second user may receive the call from the first user via the new phone number, and neither user may be made aware of the other's actual phone number. Similarly, the second user may attempt to call the first user using the new phone number shared between them. The data obscuration platform may then direct the communication to the first user such that the first user receives the call from the second user via the new phone number, and neither user may be made aware of the other's actual phone number.

[0067] Figure 3A illustrates an example 300 of communication routing between users. As shown in Figure 3A, example 300 may include a first user 302 with a first identifier 306, a second user 304 with a second identifier 308, and an identifier 320

that facilitates communication between the first user 302 and the second user 304. The first identifier 306 may be an actual identifier (e.g., actual phone number) associated with the first user 302 and the second identifier 308 may be an actual identifier (e.g., actual phone number) associated with the second user 304. The identifier 320 may be a new identifier assigned to the first and second users that is different from both the first identifier 306 and the second identifier 308. Similarly, the first user 302 may attempt to communicate with the second user 304 using the identifier 320 shared between them. The data obscuration platform may direct the communication to the second user 304 such that the second user 304 receives the communication from the first user 302 via the identifier 320, and neither user may be made aware of the other's actual identifier (e.g., the first identifier 306 and the second identifier 308).

[0068] In some embodiments, the data obscuration platform may receive requests to direct communication from a third user to the first or second user. For example, a third user may attempt to use the new phone number to contact either the first or second user. In particular, the data obscuration platform may receive, from a third user, third input that is representative of a third request to direct a communication to the first user or the second user using the third identifier. The data obscuration platform may assign, to the third user, a child identifier associated with the third identifier. This child phone number may be a derivative or spin-off of the original shared number, designed to facilitate communication while preserving the privacy of the first user's primary identifier. For example, the child identifier may be a child phone number of the third identifier (e.g., third phone number). When the third user dials the child phone number, the system intercepts the call or message and redirects it either to the first identifier of the first user or to the second identifier of the second user. The first and second users' primary phone numbers remain concealed, as the third user only interacts with the child phone number. In some embodiments, both the first user and the third user may use the child phone number to initiate communication with each other. In some embodiments, both the second user and the third user may use the child phone number to initiate

communication with each other. By dynamically generating and managing child phone numbers, the data obscuration platform ensures that all parties can communicate effectively without compromising their privacy.

[0069] The data obscuration platform may direct the communication to the first identifier of the first user or the second identifier of the second user using the child identifier. The data obscuration platform may avoid revealing the first identifier of the first user and the second identifier of the second user to the third user. To determine where to route the communication, the data obscuration platform may analyze the context and content of the communication. For instance, if the third user sends a message, natural language processing (NLP) algorithms can be used to parse the text for keywords, names, or phrases that indicate the intended recipient. Additionally, the system can leverage historical interaction data, examining past communication patterns between the third user and the first or second user to infer the likely recipient. Another method involves prompting the third user for clarification through an automated response, asking the third user to specify whom the third user wishes to contact. The system may also use metadata such as the time of day or the nature of the request to make an educated guess. For example, if the third user typically contacts the first user during business hours and the second user during personal time, this information can guide the system's decision. In some embodiments, other methods of routing communications may be used.

[0070] As an illustrative example, the above scenario may make a number semi-locked. For example, for a group of people, the data obscuration platform may create child numbers for each new participant. For instance, a first user may give an intermediary identifier to a second user. If the first user adds a third user to a group chat with the first and second users, both the first and second users may be able to see the intermediary identifier. However, the third user may see a new group chat with the intermediary number representing the first user and a new child number representing the second user. The new child number is considered a child of the intermediary number and may signify a "locked" relationship between

the second and third users. If either the first or second user calls the intermediary number, it will reach the third user, but the third user may see either the intermediary number or the new child number on their caller ID, depending on who is calling.

[0071] Figure 3B illustrates an example 350 of communication routing between users. As shown in Figure 3B, example 350 may include the first user 302 with the first identifier 306, the second user 304 with the second identifier 308, and the identifier 320 that facilitates communication between the first user 302 and the second user 304. Example 350 may also include a third user 303 who is attempting to contact either the first user 302 or the second user 304. The third user 303 may attempt to contact either the first user 302 or the second user 304 using the identifier 320. The data obscuration platform may assign, to the third user 303, a child identifier 320a associated with the third user 303. The child identifier 320a may be a derivative or spin-off of the identifier 320. In some embodiments, the data obscuration platform may generate multiple child identifiers for multiple users who attempt to contact either the first user 302 or the second user 304 using the identifier 320. The data obscuration platform may direct the communication to the first identifier 306 of the first user 302 or the second identifier 308 of the second user 304 using the child identifier 320a. The data obscuration platform may avoid revealing the first identifier 306 of the first user 302 and the second identifier 308 of the second user 304 to the third user 303.

[0072] Figure 3C illustrates an example 375 of communication routing between users. Example 375 may include the first user 302, the second user 304, a third user 305, and a fourth user 307. In some embodiments, the users may be associated with identifiers (e.g., actual phone numbers). For example, the first user 302, the second user 304, the third user 305, and the fourth user 307 may be associated with the first identifier 306, the second identifier 308, an identifier 309, and an identifier 311, respectively. In some embodiments, the data obscuration platform may assign identifier 320 to be used for communications between the first user 302 and the second user 304. Similarly, the data obscuration platform may

assign identifier 320 to be used for communications between the third user 305 and the fourth user 307. The first user 302 and the second user 304 are non-adjacent to the third user 305 and the fourth user 307, as there is no direct communication between these pairs. The data obscuration platform may ensure that identifier 320 is not assigned to adjacent pairs of users to prevent routing conflicts. This method may ensure that the same identifier is not used for directly connected users, thereby avoiding any potential overlap or confusion in routing communications. The data obscuration platform may then apply any of the methods described herein to route communications transmitted via identifier 320. In some embodiments, a different identifier (e.g., an identifier 322) may be used for adjacent or connected users (e.g., between the second user 304 and the third user 305).

[0073] In some embodiments, users may need to communicate with third-party platforms for various services, such as customer support, online transactions, or social networking. However, a significant privacy concern arises when users are required to share their personal identifiers, such as phone numbers, with these platforms. Revealing such sensitive information can lead to privacy breaches, unsolicited contact, or even identity theft. The data obscuration platform may thus route communications between the user and platform through a different, intermediary identifier. This intermediary identifier is specifically assigned to the user in connection with the third-party platform, acting as a proxy to mask the user's real phone number. When the user initiates contact with the third-party platform, the system intercepts the communication and reroutes it through the assigned identifier, ensuring that the user's actual phone number remains concealed. This approach not only protects the user's privacy but also enhances security by preventing unauthorized access to personal information. Additionally, the same intermediary identifier can be dynamically managed to connect other users to different platforms simultaneously, providing a solution for maintaining user privacy across multiple interactions.

[0074] In particular, the data obscuration platform may receive a first request to

obscure a first identifier that is associated with a user on a target platform. For example, the data obscuration platform may receive first input that is representative of a first request to obscure a first identifier that is associated with a user of a plurality of users on a target platform. As an illustrative example, the user may wish to sign up for an account on a third-party platform (e.g., a website) without revealing the user's personal data. The user may thus request that the user's personal data be obscured by the data obscuration platform. The user may submit the request through a dashboard provided by the data obscuration platform, where users can log in and submit their requests. This may allow users to specify the identifiers the users wish to obscure, such as phone numbers or email addresses, and the particular third-party platforms involved. In some embodiments, users may send an encrypted email to the data obscuration platform's support or privacy team, detailing their request and providing necessary authentication information to verify their identity. In some embodiments, users can utilize API calls from their applications, transmitting their preferences directly to the data obscuration platform. Applications associated with the data obscuration platform may also offer a privacy settings section, enabling users to easily request the obscuration of their identifiers in connection with specific third-party platforms.

[0075] The data obscuration platform may retrieve a second identifier from a database of identifiers. For example, these identifiers can be assigned to users for specific purposes, such as verifying accounts, facilitating anonymous communication, or conducting secure transactions. In some embodiments, the identifiers can be assigned to users in relation to specific relationships (e.g., with third-party platforms). In some embodiments, each identifier can be assigned to multiple users in connection with separate relationships. For example, the same identifier can be used between a first user and a first platform and, simultaneously, between a second user and a second platform. In some embodiments, identifiers may be reclaimed and reassigned to other users, ensuring efficient utilization of resources. In some embodiments, a single identifier may be reused in different scenarios by the same user at the same time.

[0076] The data obscuration platform may assign the second identifier to the user in association with the target third-party platform. In some embodiments, the second identifier is also assigned to other users of the plurality of users in associated with other third-party platforms. The data obscuration platform may efficiently manage privacy by assigning the same identifier to different relationships, such as between a first user and a first third-party platform, and a second user and a second third-party platform. This identifier, such as a phone number or email address, may act as a proxy, masking the users' actual identifiers. When the first user communicates with the first platform, the data obscuration platform routes the communication through the intermediary identifier, which the data obscuration platform recognizes and logs as part of the first relationship. Similarly, when the second user interacts with the second platform using the same intermediary identifier, the data obscuration platform logs this as a separate, distinct relationship. The data obscuration platform may maintain a detailed log of these relationships, ensuring that incoming communications are correctly routed based on the context and origin. For instance, if a message or call is received via the intermediary identifier, the data obscuration platform checks the logged relationships to determine whether the message or call pertains to the first user and first third-party platform or the second user and second third-party platform. It then routes the communication to the appropriate user or third-party platform, ensuring seamless interaction while maintaining the privacy of the users' actual identifiers. This method allows the data obscuration platform to efficiently manage multiple relationships using a single intermediary identifier, enhancing both privacy and operational efficiency.

[0077] For example, the data obscuration platform may receive, from a third party, a second request to direct a communication via the second identifier. The data obscuration platform may receive second input that is representative of a second request to direct a communication via the second identifier. The data obscuration platform may determine that the second request is associated with the target third-party platform. The data obscuration platform may then determine that

the second identifier is logged in a relationship between the target third-party platform and the first user. The data obscuration platform may then direct the communication to the first identifier that is associated with the user via the second identifier without revealing the first identifier to the third party.

[0078] In some embodiments, the data obscuration platform may remove the first identifier that is associated with the user from the target platform. For example, the user may initiate the removal request through the data obscuration platform, specifying the identifier the user wishes to remove and the target third-party platform from which it should be deleted. The data obscuration platform then communicates with the target platform, using secure channels to authenticate the request and verify the user's identity. Once authenticated, the target platform proceeds to locate and remove the user's first identifier from its records, databases, and any associated communication logs. This process may involve deleting the identifier from user profiles, contact lists, and any stored messages or call logs. The data obscuration platform ensures that the removal is comprehensive, confirming that the identifier is no longer accessible or visible on the target platform. Additionally, the data obscuration platform may provide the user with a confirmation of the removal. In some embodiments, the data obscuration platform may use any of the methods discussed below in relation to Figures 6A-6F, 7A-7E, 8, and 9 to cause removal of user data from user profiles in connection with third-party platforms.

[0079] Figure 4A illustrates an example 400 of communication routing between users and platforms. As shown in Figure 4A, example 400 may include a plurality of users, each having an actual identifier, communicating with a plurality of third-party platforms via a shared intermediary identifier. For example, user 402a, having an identifier 404a, may communicate with platform 406a via identifier 408. User 402b, having an identifier 404b, may communicate with platform 406b via identifier 408. User 402c, having an identifier 404c, may communicate with platform 406c via identifier 408. User 402d, having an identifier 404d, may communicate with platform 406d via identifier 408. In some embodiments, each of

these relationships between users and third-party platforms may be logged by the data obscuration platform to facilitate communication routing. For example, the data obscuration platform may receive a request to direct a communication via the identifier 408. The data obscuration platform may determine that the request is associated with the platform 406c. The data obscuration platform may then determine that the identifier 404c is logged in a relationship between the platform 406c and the user 402c. The data obscuration platform may then direct the communication to the identifier 404c that is associated with the user 402c via the identifier 408 without revealing the identifier 404c.

[0080] Figure 4B illustrates an example 450 of communication routing between users and platforms. In some embodiments, example 450 may include a user communicating with various platforms using various intermediary identifiers. For example, user 402a may have an actual identifier, identifier 404a. User 402a may communicate with platform 406a via identifier 410. User 402a may also communicate with platform 406b via identifier 412 and with platform 406c via identifier 414. In some embodiments, each of these relationships between user 402a and third-party platforms may be logged by the data obscuration platform to facilitate communication routing. For example, the data obscuration platform may receive a request to direct a communication via the identifier 410. The data obscuration platform may determine that the request is associated with the platform 406a. The data obscuration platform may then determine that the identifier 410 is logged in a relationship between the platform 406a and the user 402a. The data obscuration platform may then direct the communication to the identifier 404a that is associated with the user 402a via the identifier 410 without revealing the identifier 404a to the platform 406a. As another example, the data obscuration platform may receive a request to direct a communication via the identifier 412. The data obscuration platform may determine that the request is associated with the identifier 404a. The data obscuration platform may then determine that the identifier 412 is logged in a relationship between the user 402a and the platform 406b. The data obscuration platform may then direct the communication to the

platform 406b via the identifier 412 without revealing the identifier 404a to the platform 406b.

[0081] In some embodiments, the data obscuration platform may assign extensions to intermediary identifiers in associated with third-party platforms. For example, the data obscuration platform may assign, to a user, an extension for the intermediary identifier in association with the target platform. This process may involve generating a unique extension that serves as an additional layer of abstraction, ensuring that the user's identity remains protected while interacting with the target platform. The extension may be dynamically created based on specific parameters or user attributes. The data obscuration platform may then use these extensions to direct the message to the appropriate recipient. For example, the data obscuration platform may receive, from a third party, a request to direct a communication via the intermediary identifier using the extension. The data obscuration platform may determine a specific user associated with both the intermediary identifier and the extension. The data obscuration platform may then direct the communication to the proper, actual identifier via the intermediary identifier using the extension, without revealing the actual identifier to the third party.

[0082] In some embodiments, the data obscuration platform may use additional methods to route communications between platforms and users. In particular, additional methods may facilitate routing when identifiers are used simultaneously between multiple users and multiple platforms, respectively. Third-party platforms may share numbers for purposes such as one-time passwords (OTPs) or notifications via SMS or call. Therefore, the data obscuration platform may not be able to guarantee that a third-party platform's number will be unique to it. Certain logic may be used to determine who is sending a message or call, such as by analyzing the contents of the communication to determine whether an intended recipient can be inferred from the source of the communication. For example, the data obscuration platform may analyze contents of a communication to determine whether the contents include a parameter that is associated with the user. This

analysis may involve parsing the communication for specific data points or patterns that may be linked to the user's identity. These parameters may include personal identifiers such as names, email addresses, phone numbers, or other identifiers included within the contents of the communication.

[0083] In some embodiments, the data obscuration platform may determine whether the parameter that is associated with the user is derivable from the contents or from a source identifier from which the second communication originates. For example, the contents of the communication may include indirect references or contextual clues that, when pieced together, may reveal the user's identity. This may involve analyzing patterns in the text, such as specific phrases, writing styles, or recurring topics that are unique to the user. Additionally, the communication may contain metadata, such as timestamps, geolocation data, or device information, which may be cross-referenced with other data sources to infer the user's identity. Moreover, the source identifier from which the second communication originates may provide additional insights. For example, an IP address or a device ID associated with the source of the communication may be traceable back to the user through network logs or device registration records. Similarly, headers or subject lines or other protocol-specific identifiers may be used to track the origin of the communication and link it to the user. By combining these various data points, the data obscuration platform may be able to derive parameters associated with the user, even if the parameters are not explicitly stated in the communication.

[0084] In response to a determination that the contents include the parameter or that the parameter is derivable from the contents or from the source identifier, the data obscuration platform may direct the second communication to the first identifier that is associated with the user via the second identifier without revealing the first identifier to the third party. For example, the data obscuration platform may associate the parameter, included in or derived from the contents of the communication, with the user. The data obscuration platform may then direct the communication to the user without revealing the user's identifier to a third party.

[0085] Figure 5 illustrates an example 500 of communication routing using communication contents. In some embodiments, the data obscuration platform may detect a communication 510 (e.g., a message or call) via identifier 508 and originating from platform 506. The data obscuration platform may determine that multiple users are associated with identifier 508, for example, in connection with platform 506. In some embodiments, the data obscuration platform may analyze the contents of the communication 510 to determine whether the contents indicate a recipient of the communication. This may involve parsing the communication 510 for specific data points or patterns that may be linked to the user 502. These parameters may include personal identifiers such as names, email addresses, phone numbers, or other identifiers included within the contents of the communication. In some embodiments, the data obscuration platform may determine whether the parameter is derivable from the contents of the communication 510 or from a source identifier (e.g., associated with the platform 506) from which the communication 510 originates. In some embodiments, based on determining that the contents of the communication 510 include or indicate a parameter that can be linked to the user 502, the data obscuration platform may direct the communication 510 to the user 502 using the identifier 504.

[0086] In some embodiments, the data obscuration platform may determine that the contents do not include the parameter and that the parameter is not derivable from the contents or from the source identifier. The data obscuration platform may use an ML model to facilitate routing the communication to the proper user. For example, the data obscuration platform may input the contents into an ML model to cause the ML model to output a prediction as to whether the communication is intended to be directed to the user. In response to a determination that the prediction indicates that the communication is intended to be directed to the user, the data obscuration platform may direct the communication to the user.

[0087] In some embodiments, the ML model may be trained to predict users to which communications are directed based on contents of the communications. In

some embodiments, a large dataset of communications, labeled with the intended recipients, may be gathered. This dataset may include various forms of text data such as emails, chat messages, and social media posts. Data preparation may involve feature extraction, where relevant attributes from the communications are identified and quantified. These features may include lexical elements like word frequency, n-grams, and syntactic structures, as well as semantic features derived from natural language processing (NLP) techniques, such as topic modeling, sentiment analysis, and named entity recognition.

[0088] Once the features are extracted, the dataset may be split into training and testing subsets. The training subset may be used to train the ML model. In some embodiments, the ML model may take a variety of forms, some of which are discussed in greater detail in relation to Figure 9. For example, the ML model may be a supervised learning algorithm such as a support vector machine (SVM), random forest, or a deep learning model like a recurrent neural network (RNN) or transformer-based model. During training, the model may learn to associate specific patterns in the communication contents with the corresponding recipients. This learning process may involve optimizing the model's parameters to minimize prediction errors.

[0089] After training, the model's performance may be evaluated on the testing subset to ensure its accuracy and generalizability. Metrics such as precision, recall, F1-score, and accuracy may be used to assess the model's predictive capabilities. Additionally, techniques like cross-validation may be employed to further validate the model's robustness. Once validated, the trained ML model may be deployed to predict the intended recipients of new, unseen communications based on their contents. This predictive capability may enhance the efficiency and accuracy of routing communications, ensuring that messages reach their intended users.

[0090] As an illustrative example, if an incoming SMS appears to be an OTP code and identifies itself as being from a particular third-party platform, the data

obscuration platform can analyze the message content to ascertain its origin. By cross-referencing this information with the database of users who have registered with that particular third-party platform as a service, the data obscuration platform may determine which user is associated with the number. The use of AI and NLP may enable the data obscuration platform to dynamically interpret and route messages based on their content, enhancing the precision and reliability of the communication system.

[0091] In some embodiments, other methods of managing secure routing of communications may be used. For example, routing may include rule-based routing, where predefined rules and conditions determine the routing path based on specific criteria such as keywords, sender information, or message type. Additionally, peer-to-peer (P2P) networking can be utilized, where communications are routed directly between users without the need for a central server, enhancing privacy and reducing latency. Furthermore, blockchain technology offers a decentralized and secure way to route communications, leveraging smart contracts to automate and verify the routing process. In some embodiments, the data obscuration platform may use any of the methods described herein, or a combination of methods, to route communications in a way that obscures users' personal data.

[0092] In some embodiments, the data obscuration platform may purchase identifiers (e.g., phone numbers) and maintain them in a pool, ready to be issued to users based on specific use cases. This approach allows the data obscuration platform to dynamically allocate phone numbers to users as needed, ensuring that each number is optimally utilized. For instance, a user who frequently engages in business communications may be assigned a number that has been identified as suitable for professional use, while a user with more casual communication patterns may receive a different number. This method ensures that phone numbers are matched to users in a way that maximizes efficiency and relevance.

[0093] In some embodiments, the data obscuration platform may purchase

phone numbers either on-demand or in bulk, and then let these numbers “bake” for a specified period, typically ranging from 30 to 90 days. During this baking period, the data obscuration platform monitors the activity associated with each number to assess whether the number is targeted for spam communications. For example, if a number receives a high volume of calls, texts, or other communications during the baking period, it may be given a lower rating (e.g., “C” rating), indicating a higher likelihood of being flagged as spam. Conversely, a number that remains relatively quiet may be given a higher rating (e.g., “A” rating), signifying a lower risk of spam. Numbers with occasional outreach may receive a “B” rating. This rating system helps the data obscuration platform to allocate phone numbers more effectively, ensuring that high-quality numbers are reserved for users who require the most reliable communication channels.

[0094] As an illustrative example, the data obscuration platform may use these ratings to determine if a particular third-party platform is trying to contact a user. For example, the intermediary number used by the third-party platform may have an “A” rating, and the data obscuration platform may have recently assigned the intermediary number to a user. The data obscuration platform may then receive a call within a given time period, for example, five minutes after assigning the intermediary number. The data obscuration platform may predict that the call originates from that particular third-party platform. In some embodiments, the data obscuration platform may rely on user feedback to log relationships between users and other entities. For example, if the user does not correct the prediction that the call originated from the particular third-party platform, the data obscuration platform may permanently log the source of the call as the particular third-party platform.

[0095] In some embodiments, the data obscuration platform may represent each user or third-party platform as a node, with an identifier, such as a phone number, acting as a tunnel or line connecting these nodes. The goal of the data obscuration platform may be to assign phone numbers to nodes in such a way that optimizes connectivity and minimizes conflicts. For instance, two nodes (users) that frequently communicate with each other may be connected by the same

phone number, facilitating seamless interaction. This approach allows the data obscuration platform to manage and allocate phone numbers in a structured and efficient manner, ensuring that each number serves its intended purpose without unnecessary overlap or interference.

[0096] A phone number may be reused across different users and services, as well as by a single user for multiple purposes. This reuse strategy maximizes the utility of each phone number, reducing the need for constant acquisition of new numbers. For example, a phone number that was previously assigned to a user for personal communications may later be reassigned to another user for business purposes, provided that the number's history and rating make it suitable for the new use case. Similarly, a single user may use the same phone number for both personal and professional communications, leveraging the data obscuration platform's ability to manage and route calls and messages appropriately. This flexibility in phone number reuse ensures that the data obscuration platform can efficiently meet the diverse communication needs of its users while maintaining a high level of service quality.

[0097] In some embodiments, identifiers may be considered "locked," "semi-locked," or "open" or may possess another status. In this context, "locked," "semi-locked," and "open" may refer to different levels of restriction and flexibility in the use and allocation of phone numbers. A "locked" number may be assigned to a specific user for communication with a single contact, ensuring a direct and private connection where the number is exclusively used for that particular relationship. Once a number is "locked," the number may be easily reused for other users, as the "locked" relationship is known by the data obscuration platform. A "semi-locked" number, on the other hand, may allow for a bit more flexibility by enabling group communications. In this case, the number can be shared among multiple contacts within a group, but each new participant generates a unique reply number, maintaining individual privacy while facilitating group interactions. An "open" number is the most flexible, as it may be given to any set of persons and reused over time. This type of number may adapt based on usage patterns, locking

itself to the user and all interacting parties after a period of time or inactivity. Each interaction with an open number may generate a reply number, ensuring that while the number is widely accessible, individual communications remain distinct and traceable.

[0098] In some embodiments, when an intermediary phone number is issued to a user and intended for communication with another user, this intermediary number may be considered “locked.” In this scenario, the data obscuration platform may use the user’s actual number and the other user’s actual number to route calls and texts through the intermediary phone number. In some embodiments, phone numbers with a “C” rating, which indicates a higher likelihood of being flagged as spam, may be used in “locked” contexts. In contrast, numbers with “A” or “B” ratings, which are considered to have lower spam risk, may be used in any context. This rating system helps the data obscuration platform allocate phone numbers appropriately based on their quality and intended use, ensuring that high-quality numbers are reserved for more flexible and varied communication needs.

[0099] In some embodiments, “semi-locked” or “open” numbers may create a virus-like functionality that protects everyone in a given communication chain. For example a first user may be in a chat with a second user using an intermediary number and, optionally, a first child number of the intermediary number. In some embodiments, the second user may add a third user to the chat. The third user may see the second user’s real number, the intermediary number (representing the first user), and the third user’s own real number. The first user may see the intermediary number or the first child number representing the second user and a second child number of the intermediary number representing the third user. If the first user decides to add the first user’s own friend to whom that first user is talking over a third child number to the chat, the chat may have three child numbers alongside the first user. The first user’s friend who is talking over the friend’s personal number will see the third child number and new child numbers for each of the second user and the third user. The second user and third user will see a new chat with a new child number for the friend. Thus, the friend can communicate

with the third user without ever using the data obscuration platform and all relevant personal data is obscured.

[00100] In some embodiments, various scenarios may require obscuration of identifiers. The following illustrative examples may represent scenarios to which the data obscuration methods may be applied.

[00101] When signing up for a ride share service, users may be required to provide their phone numbers for login and communication purposes. To ensure privacy while allowing drivers to contact users, the data obscuration platform may generate a unique phone number for each user. This number may be consistently reachable by a rotating group of drivers who need to contact the user. Each driver may see the same intermediary number, but the user's actual phone number may remain hidden. This method ensures that the user can be contacted by different drivers without exposing the user's real phone number.

[00102] For lodging bookings, users may be required to provide their phone numbers for login and to communicate with hosts. To protect user privacy, the data obscuration platform may assign a unique phone number for each user. This number may be used to share with hosts, allowing them to contact the user without revealing the user's actual phone number. This number may be consistently reachable by a rotating group of hosts, ensuring that communication remains private and secure. This approach allows users to interact with different hosts while maintaining their privacy, as the unique number serves as an intermediary for all communications.

[00103] When shopping online, users may provide their phone numbers to receive status updates and shipping notifications. To obscure the user's real phone number, the data obscuration platform may generate a unique number for each user. This number may be used by retailers to send status updates and by shipping services to provide shipping notifications. The intermediary number may be consistently reachable by these limited, known parties, ensuring that the user's actual phone number remains private. This method allows users to receive

important updates without exposing their real phone number to multiple entities.

[00104] When visiting a business that uses an online system to manage queues, users may be required to provide their phone numbers to receive notifications. To protect user privacy, the data obscuration platform may assign a unique phone number for each user. This number may be used by the business to contact the user about the user's queue status and to call the user when the user's service is ready. The intermediary number may be consistently reachable by the business' system and staff, ensuring that the user's actual phone number remains private. This approach allows users to receive important updates from the business without exposing their real phone number.

[00105] For dating services, users may receive multiple text messages from different phone numbers, which may indicate the dating service therein. To protect user privacy, the data obscuration platform may generate unique phone numbers for each individual match. These intermediate numbers may allow users to communicate with their matches without revealing the users' actual phone numbers. Each match may see a different intermediary number, ensuring that all interactions remain private and secure. This approach allows users to mingle and communicate via dating services while maintaining their privacy and protecting their real phone numbers from exposure.

[00106] For banking services, users may be required provide their phone numbers to receive multi-factor authentication codes and other important notifications. To ensure privacy and security, the data obscuration platform may generate a unique phone number for each user. This number may be used by the bank to send multi-factor authentication codes and other messages, with variations in the bank's name (e.g., BANK1 for a first banking organization, etc.) being recognized and routed correctly. The intermediary number may be limited to only a certain number of people (e.g., 10) who can contact the user using it, ensuring that the user's actual phone number remains hidden while allowing essential communications.

[00107] In some embodiments, identifiers may take the form of email addresses. For each user and entity, the data obscuration platform may generate a unique email address using a variety of domains. These unique email addresses may ensure that communications remain private and secure. The data obscuration platform may generate child email addresses derived from the parent email address assigned to additional users. For instance, if a user generates an email address for a service on a third-party platform, it may be something like blue.green.grass@obscure.id. When support@service.com sends an email to this address, it may be routed to support.blue.green.grass@obscure.id, maintaining the obscured nature of the communication.

[00108] In some embodiments, identifiers may take other forms. For example, other types of communication identifiers, such as social media handles, instant messaging usernames, and physical addresses may be obscured using the techniques described herein. For example, for social media handles and instant messaging usernames, the data obscuration platform may generate unique identifiers for each user and service interaction. For instance, a user may be assigned a unique handle for each social media platform or messaging service, ensuring that their real identity remains hidden. Similarly, physical addresses may be managed through a system of virtual addresses or PO boxes that route mail to the user's actual location without revealing the user's address. In some embodiments, IP addresses may be masked or replaced with proxy IPs to prevent tracking and ensure the user's online activities remain anonymous. In some embodiments, any of the methods described herein may be used alone or in combination to facilitate communication routing in many contexts.

[00109] Beyond communication identifiers, other types of identifiers such as credit card numbers, SSNs, and biometric data may be obscured using the techniques described herein. For instance, the data obscuration platform may generate unique, temporary credit card numbers for each transaction, ensuring that the user's actual credit card number remains hidden. Similarly, SSNs may be replaced with unique, temporary identifiers for specific interactions with

government or financial institutions. Biometric data, such as fingerprints or facial recognition, may be encrypted and stored in a way that only generates temporary, session-specific identifiers for authentication purposes. By applying generative AI and dynamic routing, these identifiers can maintain the integrity and confidentiality of sensitive information, allowing users to interact securely across various platforms and services without exposing their true identifiers. These approaches may enhance overall data protection and reduce the risk of identity theft and fraud.

[00110] Also introduced herein is an approach that conceals or obscures users' existing credentials, such as usernames, passwords, email addresses, phone numbers, or payment cards, substituting them with altered credentials. In particular, the system may identify a user profile, on a third-party platform, that has the user's actual credentials stored thereon. The system may retrieve instructions for obscuring the user's credentials. For example, the instructions may be platform-specific such that the system is able to navigate the third-party platform to update the user's credentials. In some embodiments, the system may generate the instructions using a generative AI algorithm, for example, when the system encounters a new platform for which no existing instructions exist.

[00111] The instructions may guide the system in walking through a login process for the user using the actual credentials associated with the user. To do so, the system may encounter various authentication and verification prompts that must be satisfied in order to complete the login. The system can, in various embodiments, retrieve authentication codes from email accounts, text messages, or authentication applications associated with the user or stored on the user's devices. Once logged in, the system may replace the user's actual credentials with different credentials. For example, the system may retrieve different credentials that are obscured or otherwise secure and may update the user's profile to include the different credentials. The system may remove the actual credentials from the profile such that they are not accessible to third parties.

[00112] Figures 6A, 6B, 6C, 6D, 6E, and 6F illustrate examples of a data

obscuration platform. A data obscuration platform may provide users with a comprehensive dashboard where they can view overviews of the websites and services for which such users' data has been obscured. This dashboard may display detailed information such as the types of data obscured, the secure identifiers used, and the status of each obscuration instance. Additionally, the data obscuration platform may allow users to initiate data obscuration on new third-party platforms directly from the dashboard. By selecting a new website or service, users may trigger the generation of secure identifiers and the application of data masking techniques, ensuring their personal information remains protected. This interface may allow users to control such user's data obscuration across various online platforms.

[00113] For example, Figure 6A shows an example 600 of a data obscuration platform. Example 600 shows a dashboard on which users can view the websites and services for which such user's data has been obscured. For example, Figure 6A shows four websites or services for which the user's data has been obscured. Figure 6B shows an example 610 of a data obscuration platform. Example 610 shows a selection feature of the four websites or services for which the user's data has been obscured. For example, a selection feature may allow the user to select certain websites or services for which to view more details regarding data obscuration or for which to initiate data obscuration procedures.

[00114] Figure 6C shows an example 620 of a data obscuration platform. Example 620 shows the data obscuration platform launching or connecting to an external website of one of the third-party platforms for which the user's data has been obscured. For example, the data obscuration platform may launch a first website relating to the first platform selected in the dashboard (e.g., as shown in Figure 6B). The data obscuration platform may initiate a data obscuration process on the launched website. For example, the data obscuration platform may create different credentials for the user and may create an account for the user using the different credentials. For example, Figure 6D shows an example 630 of a data obscuration platform. Example 630 shows new payment information being added

using different data (e.g., not the user's actual data). In some embodiments, the data obscuration platform may create different credentials for the user and may replacing the user's existing credentials with the different credentials. For example, Figure 6E shows an example 640 of a data obscuration platform. Example 640 shows saved payment information in a user profile, which the data obscuration platform may replace with different data (e.g., not the user's actual data).

[00115] In some embodiments, the data obscuration platform may perform the data obscuration in various ways. Figures 7A, 7B, 7C, 7D, and 7E, discussed in greater detail below, show an illustrative example of a process of data obscuration. For example, the data obscuration platform may perform the data obscuration by guiding the user through steps to take on the website. For example, the data obscuration platform may act as a co-pilot. A co-pilot may refer to an interactive assistant or guide that helps users navigate and understand updates or modifications made to a profile on a website. This co-pilot may be an AI-driven tool or a software feature designed to provide real-time assistance and explanations as users explore the new or updated elements of the website. In some embodiments, the data obscuration platform may perform the data obscuration by performing the steps autonomously. In some embodiments, the data obscuration platform may illustrate, narrate, or otherwise show the steps being taken to the user. In some embodiments, the data obscuration platform may obscure the webpages while the steps are being taken to obscure data. In some embodiments, the data obscuration platform may use other methods of obscuring data on a website.

[00116] For example, Figure 6F shows an example 650 of a data obscuration platform. Example 650 shows a dashboard that may display, for example, detailed information such as the types of data obscured, the secure identifiers used, and the status of each obscuration instance. The dashboard may enable the user to make or initiate changes to the obscured data directly from the data obscuration platform. In some embodiments, the dashboard may enable other functionality relating to obscuration of the user's data.

[00117] Figures 7A, 7B, 7C, 7D, and 7E illustrate examples of data obscuration on a third-party platform. As discussed above, the data obscuration platform may perform the steps for data obscuration by guiding the user, acting as a co-pilot, performing the steps autonomously, or using other methods. Figure 7A shows an example 700 of an account creation page on a website associated with a third-party platform. For example, the page may feature information as well as a link to another webpage on which the user can create a new account in association with the third-party platform. Figure 7B shows an example 710 of a popup generated for display by the data obscuration platform. In some embodiments, the data obscuration platform may cause the popup to be overlayed on the website interface. In some embodiments, the data obscuration platform may cause the popup to be integrated into the website interface. The popup may, for example, prompt the user to select a button or link to initiate the generation of different credentials (e.g., not the user's actual credentials). The popup may also cause various fields to be filled with the different credentials. This may enable the user to create an account on the third-party platform's website without sharing the user's actual credentials on the website.

[00118] Figure 7C shows an example 720 of the account being generated using the different credentials. For example, the data obscuration platform may fill all relevant fields with the different credentials. In some embodiments, the data obscuration platform may also create a new profile or identity associated with the third-party platform. The new profile or identity may be stored in the data obscuration platform (e.g., as shown in Figure 6A).

[00119] Figure 7D shows an example 730 of a verification required by the third-party platform. For example, third-party platforms may have various authentication, verification, or other security measures in place. The data obscuration platform may guide the user in performing or autonomously perform the steps required to satisfy these security measures. For example, as shown in Figure 7D, the data obscuration platform may recognize that the third-party platform is prompting the user to verify the phone number entered into the user's account. In some

embodiments, the phone number may be a different phone number used to obscure the user's actual phone number. The data obscuration platform may retrieve a verification code transmitted to the phone number. In some embodiments, the data obscuration platform may generate a popup including the verification code for display on the website interface. The data obscuration platform may prompt the user to copy and paste or type the code into the relevant field. In some embodiments, the data obscuration platform may cause a relevant field to be auto-filled with the verification code. In some embodiments, the data obscuration platform may navigate other security measures prompted by a third-party platform.

[00120] Figure 7E shows an example 740 of an account update page on a website associated with a third-party platform. For example, a user may already be associated with an account on a third-party platform's website. The user may have entered actual credentials and identifiers into the user's account. The data obscuration platform may facilitate the user in updating their account to replace the user's actual data with secure data. For example, the data obscuration platform may initiate the generation of different credentials (e.g., not the user's actual credentials). The data obscuration platform may also cause various fields to be updated with the different credentials. This may enable the user to update an account on the third-party platform's website to cease sharing the user's actual credentials on the website.

[00121] In some embodiments, the data obscuration platform may retrieve a profile that includes credentials associated with a user. For example, as previously discussed, credentials may include usernames, passwords, email addresses, phone numbers, payment information, or other user data. When users create profiles on these platforms, the users may be required to provide this information to authenticate their identity and facilitate transactions. The third-party platforms may store these credentials in their databases, often employing encryption and other security measures to protect the data from unauthorized access. However, the security of these credentials can vary significantly depending on the third-party

platform's security practices and protocols. Thus, the data obscuration platform may obscure the user data in order to protect the users from breaches or other compromises to such users' data.

[00122] The data obscuration platform may retrieve a set of instructions for obscuring the credentials associated with the user on a target platform. In some embodiments, the data obscuration platform may retrieve the set of instructions from storage. For example, the data obscuration platform may locate a predetermined set of instructions stored in association with the target platform. In some embodiments, the data obscuration platform may generate the set of instructions. For example, the data obscuration platform may not have a corresponding set of instructions stored for the target platform or for the current version of the target platform. In some embodiments, the data obscuration platform may use ML models to generate the set of instructions, as discussed below in relation to Figure 9.

[00123] The set of instructions may instruct the data obscuration platform or an agent of the data obscuration platform on how to navigate the target platform in order to obscure the credentials. The instructions may specify how the data obscuration platform should authenticate itself on the website, possibly using secure tokens or temporary credentials to log in without exposing the user's actual login details. Once authenticated, the data obscuration platform may be directed to locate and access the user profile section, identifying fields that contain sensitive information such as usernames, email addresses, phone numbers, and payment details. The instructions may then outline the process for replacing or masking this data with secure, anonymized credentials, ensuring that the real information is not visible or accessible. Additionally, the data obscuration platform may be instructed to update any linked accounts or services to reflect these changes, maintaining consistency across the user's digital footprint. Finally, the data obscuration platform may be directed to log out and clear any session data to prevent unauthorized access.

[00124] Based on the set of instructions, the data obscuration platform may perform a login on behalf of the user on the target platform using at least one of the credentials associated with the user. The data obscuration platform may navigate the target platform's website by executing a series of steps outlined in the set of instructions. In some embodiments, the data obscuration platform may employ web scraping techniques or utilize the target platform's application programming interface (API) to programmatically access a user profile section. The data obscuration platform may identify and locate specific HTML elements that contain sensitive user data, such as input fields for usernames, email addresses, phone numbers, and payment information.

[00125] In some embodiments, performing a login on behalf of a user involves satisfying authentication, verification, or other security measures. Third-party platforms' websites may require users to undergo various authentication, verification, and security measures to ensure the integrity and security of their accounts and data. Authentication processes may include multi-factor authentication (MFA), where users must provide two or more verification factors, such as a password and a one-time code sent to their mobile device, to gain access. Verification measures may involve identity verification steps, such as submitting government-issued identification or answering security questions to confirm the user's identity. Additionally, third-party platforms may implement security measures like CAPTCHA tests to distinguish between human users and automated bots, and employ encryption protocols to protect data transmission. These measures may aim to safeguard user accounts from unauthorized access, prevent fraudulent activities, and ensure that only legitimate users can interact with the third-party platform's services.

[00126] As an illustrative example, the data obscuration platform may identify a prompt, by the target platform, for an authentication code. For example, the authentication code may be transmitted to a separate account associated with the user. In some embodiments, the authentication code may be transmitted to an email address, a phone number, an application, or another identifier associated

with the user. In some embodiments, the authentication code may be transmitted to a different identifier (e.g., not the user's actual account or identifier) that has been generated by the data obscuration platform. In some embodiments, the data obscuration platform may retrieve the authentication code from the separate account associated with the user. For example, the data obscuration platform may receive the code directly if the account or identifier is managed by the data obscuration platform. In some embodiments, the data obscuration platform may receive or may have previously received the user's credentials associated with the account or identifier to which the code was transmitted. The data obscuration platform may perform a login in connection with the account or identifier to retrieve the authentication code. In some embodiments, the data obscuration platform may prompt the user to retrieve the code, for example, if the data obscuration platform is unable to retrieve the code.

[00127] The data obscuration platform may submit the authentication code to the target platform. The data obscuration platform may act as an intermediary to enhance security and privacy for the user. By handling the authentication process, the data obscuration platform may ensure that the user's actual credentials are never exposed to the target platform. In some embodiments, the data obscuration platform may employ encryption techniques to protect the authentication code during transmission, further safeguarding the user's information from potential interception or unauthorized access.

[00128] The data obscuration platform may generate different credentials for the user. The data obscuration platform may employ a variety of techniques designed to enhance security and privacy. As an illustrative example, for email addresses, the data obscuration platform may create email addresses for each service the user interacts with. These email addresses may be generated using a combination of random characters and domain names controlled by the data obscuration platform, such as "user1234@obscuremail.com." This approach ensures that the user's real email address remains hidden, and any emails sent to the different address can be forwarded to the user's actual inbox or managed within the data

obscuration platform. If the different email address is compromised or no longer needed, it can be easily deactivated without affecting the user's primary email account.

[00129] As an illustrative example, the data obscuration platform may generate strong, unique passwords for each service the user accesses. These passwords may be created using a combination of uppercase and lowercase letters, numbers, and special characters, ensuring they meet the complexity requirements of most websites. For example, a generated password may look like "A1b2C3d4!@#". The data obscuration platform may store these passwords securely using encryption and provide them to the user through a password manager interface (e.g., as shown in Figure 6F). This method not only enhances security by preventing password reuse across multiple sites but also simplifies the user's experience by managing complex passwords on their behalf.

[00130] As an illustrative example, when it comes to payment information, the data obscuration platform may generate virtual credit card numbers for each transaction. These virtual cards may be linked to the user's actual credit card but have unique numbers, expiration dates, and CVV codes. For instance, a virtual card number may be "4111 1111 1111 1111" with an expiration date of "12/25" and a CVV of "123." The data obscuration platform may set limits on these virtual cards, such as a maximum transaction amount or a specific expiration date, to further enhance security. By using virtual credit cards, the user's real payment information remains protected, and any fraudulent activity can be quickly mitigated by deactivating the compromised virtual card.

[00131] Overall, the data obscuration platform's ability to generate different credentials significantly enhances the user's security and privacy. By creating different credentials, the data obscuration platform ensures that the user's real information is never exposed, reducing the risk of data breaches and unauthorized access.

[00132] The data obscuration platform may submit the different credentials to

the target platform. The instructions may direct the data obscuration platform to replace the content of these fields with anonymized tokens or encrypted placeholders, ensuring that the original data is obscured. Additionally, the data obscuration platform may update any associated metadata or linked services to maintain data consistency. Upon completion, the data obscuration platform may execute a logout sequence and clear any session cookies or local storage data to prevent unauthorized access, thereby ensuring the secure and effective obscuration of user data on the target platform.

[00133] The data obscuration platform may perform the login on behalf of the user on the target platform using the different credentials associated with the user. The data obscuration platform may utilize the different credentials it has generated to ensure the user's real information remains protected. For instance, the data obscuration platform may use a disposable email address, such as "user1234@obscuremail.com," and a strong, unique password like "A1b2C3d4!@#" to authenticate the user on the target website. The data obscuration platform may programmatically fill in the login form fields with these credentials, leveraging secure autofill techniques to prevent exposure of the data.

[00134] The data obscuration platform may update the profile to include the different credentials. Upon accessing the profile settings, the data obscuration platform may systematically replace the existing sensitive information with the new, anonymized credentials. For example, the data obscuration platform may update the email address field with a disposable email like "user1234@obscuremail.com" and change the password to a strong, unique one such as "A1b2C3d4!@#". Additionally, if the profile includes payment information, the data obscuration platform may substitute the real credit card details with a virtual card number, expiration date, and CVV code, such as "4111 1111 1111 1111," "12/25," and "123," respectively. Throughout this process, the data obscuration platform may ensure that all data transmissions are encrypted to prevent unauthorized access. By updating the profile with these different credentials, the data obscuration platform effectively obscures the user's real information, reducing the risk of data

breaches and unauthorized access while maintaining the functionality and usability of the user's account.

[00135] In some embodiments, the data obscuration platform may submit, to the target platform, a removal request for removal of the credentials associated with the user from the target platform. For example, the data obscuration platform may request that the target platform update or clear personal information, such as email addresses, passwords, and other user data. Certain third-party platforms may allow users to change or anonymize personal details without needing to delete the account. The data obscuration platform may also request that the third-party platform disable or revoke access to connected services, like social media or third-party logins, and clear stored data such as saved preferences or browsing history. Additionally, some platforms provide the option to limit data sharing or tracking features, which can further reduce the amount of personal information associated with the account.

[00136] Figure 8 illustrates a process 800 for obscuring credentials associated with users. Process 800 represents an overview of certain methods of obscuring credentials, though other methods may be used. In some embodiments, a combination of methods can be used to obscure credentials.

[00137] As shown in Figure 8, the process 800 may be triggered when a user wishes to obscure such user's data. A user may indicate to a data obscuration platform their desire to obscure their user data on third-party platforms through, for example, a user interface. This may begin with the user accessing the data obscuration platform's management dashboard, where they can find options for data privacy and management. The user may then select the specific third-party platforms on which the user wishes to obscure such user's data, such as social media accounts, e-commerce sites, or online services. By providing the necessary account details, the user enables the data obscuration platform to take action. In some embodiments, the process 800 may be triggered by a breach or information leak. For example, the data obscuration platform may receive a notification from

another system or entity indicating a breach or information leak on certain third-party platforms in which the user may be implicated. In some embodiments, the process 800 may be triggered by another objective.

[00138] The data obscuration platform's management dashboard may prompt a driver agent to take certain actions in response to the user initiating the data obscuration process. In some embodiments, the driver may be a browser extension, a desktop application, a mobile application, or another form of driver. A driver agent may act as an intermediary that facilitates communication between hardware devices and software applications. This agent may be responsible for translating high-level commands from software into low-level instructions that the hardware can execute, ensuring seamless operation and integration. For instance, the driver agent may manage the interaction between the data obscuration platform and peripheral systems such as third-party platforms. Additionally, the driver agent may provide diagnostic and troubleshooting capabilities, identifying and resolving issues that may arise during operation.

[00139] In some embodiments, the driver agent may access the user's data that is to be encrypted. In some embodiments, the driver agent may obtain a secure key-pair to access encrypted user data, ensuring that the data remains protected while enabling necessary operations. This process may begin with the driver agent securely connecting to a key management service (KMS) or a hardware security module (HSM) where the key-pair is stored. The driver agent may authenticate itself using credentials or tokens to gain access to the key management system. Once authenticated, the driver agent may request the retrieval of the public-private key pair, which is then securely transmitted to the agent using encryption protocols to prevent interception. The public key may be used to encrypt data, while the private key, which remains securely stored within the driver agent's environment, may be used to decrypt the user data when needed. To further enhance security, the driver agent may implement additional measures such as secure key storage solutions and multi-factor authentication to protect the private key from unauthorized access.

[00140] In some embodiments, the driver agent may identify an API to assist in accessing encrypted user data. For example, the driver agent may identify the appropriate API by reviewing the documentation provided by the encryption service or library, which may include details on available functions, authentication methods, and usage examples. Once the suitable API is identified, the driver agent may authenticate itself using credentials such as API keys, tokens, or certificates to establish a secure connection with the encryption service. After successful authentication, the driver agent may call the relevant API endpoints to perform encryption operations. This may involve sending the user data to the API in a secure manner to ensure data integrity and confidentiality during transmission. The API may then process the data and return the encrypted output, which the driver agent may store or transmit as needed.

[00141] In some embodiments, information regarding a given third-party platform may be input into the data obscuration platform. For example, the third-party platform may host a website that was involved in a recent data breach involving the user. In some embodiments, the third-party platform may host a website on which the user wishes to obscure such user's data. In some embodiments, the data obscuration platform may utilize an executor agent to interpret the third-party platform information. For example, the executor agent may begin by analyzing the website's Document Object Model (DOM). The DOM is a hierarchical representation of the HTML elements that make up a webpage, including tags, attributes, and content. By accessing the DOM, the executor agent can systematically traverse the structure of the webpage, identifying key elements such as forms, buttons, links, and input fields that are essential for navigation and interaction.

[00142] To focus on the relevant parts of the webpage, the executor agent may filter out extraneous styling and scripts. This involves removing CSS (Cascading Style Sheets) and JavaScript code that are not necessary for understanding the core functionality and content of the page. By stripping away these elements, the agent can isolate the body and metadata of the HTML, which contain the primary

information and interactive components. This cleaned DOM provides a streamlined view of the webpage, making it easier for the agent to identify and interpret the elements that are crucial for generating navigation instructions. In some embodiments, the executor agent may use various methods to interpret the cleaned DOM. For example, the executor agent may use NLP to determine semantic meaning to elements of the webpage.

[00143] Once the cleaned DOM is extracted, the executor agent can parse the website to interpret the webpages in order. Parsing involves breaking down the HTML into its constituent parts and understanding the relationships between them. The agent may analyze the hierarchical structure of the DOM, noting the parent-child relationships between elements and the sequence in which they appear. This allows the agent to map out the logical flow of the webpage, such as the sequence of steps required to navigate from the homepage to a specific settings page where user credentials can be changed.

[00144] During the parsing process, the executor agent may also identify interactive elements and their associated actions. For example, it may recognize a “Submit” button within a form and understand that clicking this button triggers a form submission. Similarly, the agent may detect input fields for entering new credentials and links that lead to different sections of the website. By understanding these interactions, the agent can generate a set of instructions that detail the precise actions needed to navigate the website and perform tasks such as changing user credentials to obscure data.

[00145] In addition to generating navigation instructions, the executor agent may also incorporate error handling and validation steps. For instance, it may include instructions for verifying that the new credentials have been successfully updated or for handling potential errors such as incorrect password entries. By thoroughly interpreting the website’s DOM and parsing the HTML content, the executor agent ensures that the generated instructions are accurate, reliable, and capable of guiding the user or an automated system through the necessary steps to achieve

the desired outcome.

[00146] In some embodiments, as discussed in greater detail below, the data obscuration platform may receive screenshots of the third-party platform. In some embodiments, the data obscuration platform may use a computer vision model to interpret the screenshots and facilitate generation of instructions for navigating the third-party platform. The executor agent or the computer vision model may pass the information to another agent trained to determine high-level objectives.

[00147] For example, the data obscuration platform may initiate an objective workflow to determine a high-level objective associated with the obscuration of the user's data. To do so, the data obscuration platform may request the creation of a planner agent based on objective-based workflows. This planner agent may be designed to handle various tasks such as identifying sensitive data, determining the appropriate methods for data obscuration, and executing these methods in a systematic manner. By leveraging objective-based workflows, the planner agent may ensure that each step in the data obscuration process aligns with predefined goals and criteria, thereby enhancing efficiency and accuracy. For instance, the planner agent may prioritize tasks based on the sensitivity of the data, the compliance requirements of different jurisdictions, and the specific preferences of the user. This structured approach may help in minimizing errors and ensuring that the data obscuration process is both comprehensive and consistent.

[00148] In some embodiments, an objective may be a specific goal or outcome that the planner agent aims to achieve as part of the data obscuration process. For instance, an objective may be to ensure the anonymization of personal data on a given platform. This objective may involve identifying and obscuring user data within datasets to prevent unauthorized access or misuse. Another objective may be to enhance data security by applying encryption techniques to sensitive financial records, ensuring that only authorized users can access the decrypted information. Additionally, an objective may focus on maintaining data integrity and accuracy while performing obscuration tasks, ensuring that the data remains useful

for legitimate purposes such as analytics or reporting.

[00149] In some embodiments, the data obscuration platform may request that a generative AI algorithm create the planner agent. This approach may offer several advantages, including the ability to dynamically generate and adapt the planner agent based on real-time data and evolving requirements. The generative AI algorithm may analyze the current data landscape, user preferences, and regulatory guidelines to create a customized planner agent that is tailored to the specific needs of the user. This may involve generating workflows that are optimized for different types of data, such as personal information, financial records, or medical data, and selecting the most effective obscuration techniques for each type. By utilizing generative AI, the data obscuration platform may ensure that the planner agent is not only effective but also adaptable to changing conditions and requirements.

[00150] Furthermore, the use of a generative AI algorithm to create the planner agent may enhance the data obscuration platform's ability to handle complex and large-scale data obscuration tasks. The AI algorithm may employ advanced ML techniques (e.g., as discussed in greater detail in relation to Figure 9) to continuously improve the planner agent's performance, learning from past experiences and user feedback. This may result in more efficient workflows, reduced processing times, and higher levels of data security. Additionally, the generative AI may enable the planner agent to identify and address potential vulnerabilities or inefficiencies in the data obscuration process, thereby enhancing the overall robustness of the data obscuration platform.

[00151] In particular, the data obscuration platform may request that a generative AI create the planner agent by initiating a series of steps involving API integration, data transmission, and response handling. Initially, the data obscuration platform may integrate with the generative AI service by obtaining an API key or token and using it to authenticate requests. This authentication may involve including the key in the HTTP headers or as a query parameter. Next, the

data obscuration platform may define a request payload that specifies the requirements and parameters for the planner agent, including objective-based workflows, data characteristics, compliance requirements, and user preferences. The data obscuration platform may then make an API call to the generative AI service, targeting the specific endpoint provided for creating planner agents. Upon receiving the request, the generative AI service may process the information and generate the planner agent, returning a response from which the data obscuration platform may extract relevant information about the created planner agent. The data obscuration platform may also manage any errors or exceptions returned by the API, such as invalid parameters or authentication failures. Once the planner agent is created, the data obscuration platform may save its details, integrate it into the existing architecture, and initiate it to start generating objectives. Throughout this process, the data obscuration platform may continuously monitor the performance of the planner agent and provide feedback to the generative AI service for further optimization, ensuring that the data obscuration process is both efficient and compliant with relevant standards.

[00152] The planner agent may generate a proposed workflow with sequences of steps for the driver agent to execute. This workflow may be designed to address specific objectives, such as data obscuration, encryption, or compliance with regulatory standards. For instance, the planner agent may outline a series of steps starting with the identification of sensitive data, followed by the obscuration techniques, and concluding with the verification of data integrity. Each step in the workflow may include detailed instructions and criteria for the driver to follow, such as the algorithms to use, the data formats to handle, and the security protocols to implement. By providing a clear and organized sequence of actions, the planner agent may help the driver execute tasks systematically, reducing the likelihood of errors and enhancing overall efficiency. Additionally, the proposed workflow may be adaptable, allowing the driver to adjust steps based on real-time feedback or changing requirements, thereby ensuring that the process remains aligned with the desired outcomes and standards. The planner agent may then transmit the

steps to the driver agent for execution.

[00153] As an illustrative example, a set of instructions may include: 1. Navigate to website; 2. Determine if user is logged in or logged out; 3. If logged in, log the user out; 4. Navigate to login page or determine if login selectors are on page; 5. Execute login; 6. Navigate to {objective}; 7. Within {objective} determine if user information is needed or generated information is needed; and 8. If unable to perform any step of the sequence, try navigating to an alternative path, and if no next step is generated, request user input.

[00154] In some embodiments, the driver agent may be a browser extension. As a browser extension, the driver agent may integrate directly with the user's web browser, allowing it to interact with web pages in real-time. It may monitor the user's activities, identify sensitive data, and automatically perform actions such as changing credentials, deleting cookies, or filling forms with anonymized information. The extension may also provide a user interface for manual interventions, enabling users to approve or modify actions as needed. As a desktop application, the driver agent may operate independently of the browser, offering functionality across different applications and systems. It may scan local files, manage system settings, and interact with various software to obscure data comprehensively. The desktop application may also include features like encrypted storage for sensitive information and automated workflows for regular data privacy maintenance. In some embodiments, the driver agent may take other forms, such as a mobile application.

[00155] The driver agent may initiate the workflow and begin executing the given steps. For example, each step in the workflow may come with specific instructions, parameters, and criteria that the driver agent must adhere to. For instance, if the workflow involves data encryption, the first step may be to identify and retrieve the user data (e.g., user credentials) from a specified database. The driver agent may then authenticate itself to access the database, ensuring that it has the necessary permissions to retrieve the data securely. Once the data is retrieved, the driver

agent proceeds to the next step, which may involve generating different credentials according to the specified parameters. After completing the encryption, the driver agent may store the different credentials in a secure location, as outlined in the workflow.

[00156] In some embodiments, the driver agent may extract or detail information that a third-party platform may have about a user by initially pulling all pertinent data from the user's account (e.g., a LinkedIn account), such as posts, connections, or personally identifiable information (PII). This extraction process may involve gathering extensive data to form a comprehensive profile of the user. After the data is extracted, the driver agent may securely store this information, linking it to the user. Following this, the driver agent may assist in cleaning the user's LinkedIn data by obscuring or removing specific data points as per the user's request.

[00157] After executing each step, the driver agent may report back to the planner agent, providing detailed feedback on the execution status. This feedback may include information such as the completion status of the step, any errors or issues encountered, and the results of the operation. For example, after generating different credentials, the driver agent may report back to the planner agent with a status update indicating that the credential generation was successful, along with any relevant metadata. If any errors or issues were encountered, the driver agent may provide detailed error messages and logs to help the planner agent diagnose and address the problem.

[00158] The planner agent may use this feedback to monitor the progress of the workflow and make any necessary adjustments. For instance, if the driver agent reports an error during the encryption step, the planner agent may modify the workflow to include additional error handling or retry mechanisms. In some embodiments, the planner agent may be required to rebuild instructions to address failed operations. Additionally, the planner agent may use the feedback to optimize future workflows, learning from past experiences to improve efficiency and

reliability. By continuously reporting back to the planner agent, the driver agent ensures that the workflow is executed accurately and efficiently, while also providing valuable insights that can be used to enhance the overall process.

[00159] In some embodiments, if human input is required during the execution of a workflow, the driver agent may create a co-pilot request to the user, ensuring that the necessary human intervention is integrated into the workflow. This may occur at various stages of the workflow where automated decision-making may not be sufficient or where user authorization is needed. For instance, if the workflow involves approving access to sensitive data, the driver agent may pause its operations and generate a co-pilot request, prompting the user to review and approve the access request. This request may be delivered through an interface, such as a notification in a dashboard or an email alert, providing the user with all relevant details and context to make a decision. The user may then provide their input, such as granting or denying access, adding comments, or selecting from predefined options. Once the user's input is received, the driver agent may resume the workflow, incorporating the user's decision into the subsequent steps.

[00160] Once all steps of the objective-based workflow are completed, the objective may be deemed complete. In some embodiments, the driver agent performs a final review to ensure that the objective has been met according to the predefined criteria. This may include confirming that all user data has been obscured. The driver agent reports the successful completion of the objective back to the planner agent, providing a comprehensive summary of the operations performed and the results achieved. By systematically executing and validating each step, the driver agent ensures that the objective is completed efficiently, accurately, and securely, thereby fulfilling the intended purpose of the workflow.

[00161] In some embodiments, the data obscuration platform may use batch processing to perform any of the above steps of process 800. For example, the system may be designed to retrieve a comprehensive list of websites where the user has accounts and systematically perform a series of predefined steps for each

site. Initially, the system may gather the list of websites from a secure database or user-provided input. For each website, the system initiates a session to access the site, using stored credentials or prompting the user for authentication if necessary. Once logged in, the system navigates to the relevant sections of the website, such as account settings or privacy controls, by interpreting the website's DOM and following the generated navigation instructions. It then performs the required actions to obscure the user's data, such as changing passwords, updating email addresses, or deleting personal information. After completing the data obscuration tasks, the system logs out and moves on to the next website in the list. Throughout this process, the system maintains detailed logs of the actions performed and any issues encountered, ensuring that each step is documented and any errors can be addressed promptly. By automating these tasks in a batch processing manner, the system enhances efficiency and ensures consistent application of data privacy measures across multiple websites.

[00162] Figure 9 includes an illustration 900 of inputs and outputs of a model 904. In some embodiments, the model 904 may be integrated with or may work in conjunction with a planner agent, as described above in relation to Figure 8. In some embodiments, the model 904 may generate a set of instructions for a driver agent to execute in carrying out an objective. In some embodiments, the model 904 may carry out other functions, as well.

[00163] A "model," as used herein, can refer to a construct that is trained using training data to make predictions or provide probabilities for new data items, whether or not the new data items were included in the training data. For example, training data for supervised learning can include items with various parameters and an assigned classification. A new data item can have parameters that a model can use to assign a classification to the new data item. As another example, a model can be a probability distribution resulting from the analysis of training data, such as a likelihood of an n-gram occurring in a given language based on an analysis of a large corpus from that language. Examples of models include neural networks, support vector machines, decision trees, Parzen windows, Bayes,

clustering, reinforcement learning, probability distributions, decision trees, decision tree forests, and others. Models can be configured for various situations, data types, sources, and output formats.

[00164] In some implementations, the model 904 can be a neural network with multiple input nodes that receive, for example, information about third-party platforms. The input nodes can correspond to functions that receive the input and produce results. These results can be provided to one or more levels of intermediate nodes that each produce further results based on a combination of lower-level node results. A weighting factor can be applied to the output of each node before the result is passed to the next layer node. At a final layer, ("the output layer") one or more nodes can produce a value classifying the input. In some implementations, such neural networks, known as deep neural networks, can have multiple layers of intermediate nodes with different configurations, can be a combination of models that receive different parts of the input and/or input from other parts of the deep neural network, or are convolutions, partially using output from previous iterations of applying the model as further input to produce results for the current input.

[00165] An ML model can be trained with supervised learning, where the training data includes platform website information or images as input and a desired output, such as a set of instructions for navigating the website to update a user profile. A representation of the third-party platform's website can be provided to the model. Output from the model can be compared to the desired output for that website and, based on the comparison, the model can be modified, such as by changing weights between nodes of the neural network or parameters of the functions used at each node in the neural network (e.g., applying a loss function). After modifying the model in this manner, the model can be trained to evaluate new platforms.

[00166] To assist in understanding the present disclosure, some concepts relevant to neural networks and ML are discussed herein. Generally, an ML model can include any prediction model or other statistical model used to predict a result

based on an input based on examples. An ML model may include an artificial intelligence model (e.g., a large language model (LLM), a deep learning model, a generative artificial intelligence model, and the like) as well as classification models or other types of models. Thus, persons of ordinary skill in the art will recognize that, unless specifically stated, any instance of an ML model, artificial intelligence model, or classification model may be switched with another ML model, artificial intelligence model, or classification model, and the present disclosure is not to be construed as limited to the examples recited herein.

[00167] Generally, a neural network includes a number of computation units (sometimes referred to as “neurons”). Each neuron receives an input value and applies a function to the input to generate an output value. The function typically includes a parameter (also referred to as a “weight”) whose value is learned through the process of training. A plurality of neurons may be organized into a neural network layer (or simply “layer”) and there may be multiple such layers in a neural network. The output of one layer may be provided as input to a subsequent layer. Thus, input to a neural network may be processed through a succession of layers until an output of the neural network is generated by a final layer. This is a simplistic discussion of neural networks and there may be more complex neural network designs that include feedback connections, skip connections, and/or other such possible connections between neurons and/or layers, which are not discussed in detail here.

[00168] A deep neural network (DNN) is a type of neural network having multiple layers and/or a large number of neurons. The term DNN may encompass any neural network having multiple layers, including convolutional neural networks (CNNs), RNNs, multilayer perceptrons (MLPs), Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Auto-regressive Models, among others.

[00169] DNNs are often used as ML-based models for modeling complex behaviors (e.g., human language, image recognition, object classification) in order

to improve the accuracy of outputs (e.g., more accurate predictions) such as, for example, as compared with models with fewer layers. In the present disclosure, the term “ML-based model” or more simply “ML model” may be understood to refer to a DNN. Training an ML model refers to a process of learning the values of the parameters (or weights) of the neurons in the layers such that the ML model is able to model the target behavior to a desired degree of accuracy. Training typically requires the use of a training dataset, which is a set of data that is relevant to the target behavior of the ML model.

[00170] As an example, to train an ML model that is intended to model human language (also referred to as a language model), the training dataset may be a collection of text documents, referred to as a text corpus (or simply referred to as a corpus). The corpus may represent a language domain (e.g., a single language), a subject domain (e.g., scientific papers), and/or may encompass another domain or domains, be they larger or smaller than a single language or subject domain. For example, a relatively large, multilingual and non-subject-specific corpus may be created by extracting text from online webpages and/or publicly available social media posts. Training data may be annotated with ground truth labels (e.g., each data entry in the training dataset may be paired with a label), or may be unlabeled.

[00171] Training an ML model generally involves inputting into an ML model (e.g., an untrained ML model) training data to be processed by the ML model, processing the training data using the ML model, collecting the output generated by the ML model (e.g., based on the inputted training data), and comparing the output to a desired set of target values. If the training data is labeled, the desired target values may be, e.g., the ground truth labels of the training data. If the training data is unlabeled, the desired target value may be a reconstructed (or otherwise processed) version of the corresponding ML model input (e.g., in the case of an autoencoder), or can be a measure of some target observable effect on the environment (e.g., in the case of a reinforcement learning agent). The parameters of the ML model are updated based on a difference between the generated output value and the desired target value. For example, if the value outputted by the ML

model is excessively high, the parameters may be adjusted so as to lower the output value in future training iterations. An objective function is a way to quantitatively represent how close the output value is to the target value. An objective function represents a quantity (or one or more quantities) to be optimized (e.g., minimize a loss or maximize a reward) in order to bring the output value as close to the target value as possible. The goal of training the ML model typically is to minimize a loss function or maximize a reward function.

[00172] The training data may be a subset of a larger data set. For example, a data set may be split into three mutually exclusive subsets: a training set, a validation (or cross-validation) set, and a testing set. The three subsets of data may be used sequentially during ML model training. For example, the training set may be first used to train one or more ML models, each ML model, e.g., having a particular architecture, having a particular training procedure, being describable by a set of model hyperparameters, and/or otherwise being varied from the other of the one or more ML models. The validation (or cross-validation) set may then be used as input data into the trained ML models to, e.g., measure the performance of the trained ML models and/or compare performance between them. Where hyperparameters are used, a new set of hyperparameters may be determined based on the measured performance of one or more of the trained ML models, and the first step of training (i.e., with the training set) may begin again on a different ML model described by the new set of determined hyperparameters. In this way, these steps may be repeated to produce a more performant trained ML model. Once such a trained ML model is obtained (e.g., after the hyperparameters have been adjusted to achieve a desired level of performance), a third step of collecting the output generated by the trained ML model applied to the third subset (the testing set) may begin. The output generated from the testing set may be compared with the corresponding desired target values to give a final assessment of the trained ML model's accuracy. Other segmentations of the larger data set and/or schemes for using the segments for training one or more ML models are possible.

[00173] Backpropagation is an algorithm for training an ML model. Backpropagation is used to adjust (also referred to as update) the value of the parameters in the ML model, with the goal of optimizing the objective function. For example, a defined loss function is calculated by forward propagation of an input to obtain an output of the ML model and a comparison of the output value with the target value. Backpropagation calculates a gradient of the loss function with respect to the parameters of the ML model, and a gradient algorithm (e.g., gradient descent) is used to update (i.e., “learn”) the parameters to reduce the loss function. Backpropagation is performed iteratively so that the loss function is converged or minimized. Other techniques for learning the parameters of the ML model may be used. The process of updating (or learning) the parameters over many iterations is referred to as training. Training may be carried out iteratively until a convergence condition is met (e.g., a predefined maximum number of iterations has been performed, or the value outputted by the ML model is sufficiently converged with the desired target value), after which the ML model is considered to be sufficiently trained. The values of the learned parameters may then be fixed and the ML model may be deployed to generate output in real-world applications (also referred to as “inference”).

[00174] In some examples, a trained ML model may be fine-tuned, meaning that the values of the learned parameters may be adjusted slightly in order for the ML model to better model a specific task. Fine-tuning of an ML model typically involves further training the ML model on a number of data samples (which may be smaller in number/cardinality than those used to train the model initially) that closely target the specific task. For example, an ML model for generating natural language that has been trained generically on publicly-available text corpora may be, e.g., fine-tuned by further training using specific training samples. The specific training samples can be used to generate language in a certain style or in a certain format. For example, the ML model can be trained to generate a blog post having a particular style and structure with a given topic.

[00175] Some concepts in ML-based language models are now discussed. It may be noted that, while the term “language model” has been commonly used to refer to an ML-based language model, there may exist non-ML language models. In the present disclosure, the term “language model” may be used as shorthand for an ML-based language model (i.e., a language model that is implemented using a neural network or other ML architecture), unless stated otherwise. For example, unless stated otherwise, the “language model” encompasses LLMs.

[00176] A language model may use a neural network (typically a DNN) to perform NLP tasks. A language model may be trained to model how words relate to each other in a textual sequence, based on probabilities. A language model may contain hundreds of thousands of learned parameters or in the case of an LLM may contain millions or billions of learned parameters or more. As non-limiting examples, a language model can generate text, translate text, summarize text, answer questions, write code (e.g., Python, JavaScript, or other programming languages), classify text (e.g., to identify spam emails), create content for various purposes (e.g., social media content, factual content, or marketing content), or create personalized content for a particular individual or group of individuals. Language models can also be used for chatbots (e.g., virtual assistance).

[00177] In recent years, there has been interest in a type of neural network architecture, referred to as a transformer, for use as language models. For example, the Bidirectional Encoder Representations from Transformers (BERT) model, the Transformer-XL model, and the Generative Pre-trained Transformer (GPT) models are types of transformers. A transformer is a type of neural network architecture that uses self-attention mechanisms in order to generate predicted output based on input data that has some sequential meaning (i.e., the order of the input data is meaningful, which is the case for most text input). Although transformer-based language models are described herein, it should be understood that the present disclosure may be applicable to any ML-based language model, including language models based on other neural network architectures such as RNN-based language models.

[00178] Existing language models include language models that are based only on the encoder of the transformer or only on the decoder of the transformer. An encoder-only language model encodes the input text sequence into feature vectors that can then be further processed by a task-specific layer (e.g., a classification layer). BERT is an example of a language model that can be considered to be an encoder-only language model. A decoder-only language model accepts embeddings as input and can use auto-regression to generate an output text sequence. Transformer-XL and GPT-type models can be language models that are considered to be decoder-only language models.

[00179] Because GPT-type language models tend to have a large number of parameters, these language models can be considered LLMs. An example of a GPT-type LLM is GPT-3. GPT-3 is a type of GPT language model that has been trained (in an unsupervised manner) on a large corpus derived from documents available to the public online. GPT-3 has a very large number of learned parameters (on the order of hundreds of billions), is able to accept a large number of tokens as input (e.g., up to 2,048 input tokens), and is able to generate a large number of tokens as output (e.g., up to 2,048 tokens). GPT-3 has been trained as a generative model, meaning that it can process input text sequences to predictively generate a meaningful output text sequence. ChatGPT is built on top of a GPT-type LLM and has been fine-tuned with training datasets based on text-based chats (e.g., chatbot conversations). ChatGPT is designed for processing natural language, receiving chat-like inputs, and generating chat-like outputs.

[00180] A computer system can access a remote language model (e.g., a cloud-based language model), such as ChatGPT or GPT-3, via a software interface (e.g., an API). Additionally or alternatively, such a remote language model can be accessed via a network such as, for example, the Internet. In some implementations, such as, for example, potentially in the case of a cloud-based language model, a remote language model can be hosted by a computer system that can include a plurality of cooperating (e.g., cooperating via a network) computer systems that can be in, for example, a distributed arrangement. Notably,

a remote language model can employ a plurality of processors (e.g., hardware processors such as, for example, processors of cooperating computer systems). Indeed, processing of inputs by an LLM can be computationally expensive/can involve a large number of operations (e.g., many instructions can be executed/large data structures can be accessed from memory), and providing output in a required timeframe (e.g., real time or near real time) can require the use of a plurality of processors/cooperating computing devices as discussed above.

[00181] Inputs to an LLM can be referred to as a prompt, which is a natural language input that includes instructions to the LLM to generate a desired output. A computer system can generate a prompt that is provided as input to the LLM via its API. As described above, the prompt can optionally be processed or pre-processed into a token sequence prior to being provided as input to the LLM via its API. A prompt can include one or more examples of the desired output, which provides the LLM with additional information to enable the LLM to generate output according to the desired output. Additionally or alternatively, the examples included in a prompt can provide inputs (e.g., example inputs) corresponding to/as can be expected to result in the desired outputs provided. A one-shot prompt refers to a prompt that includes one example, and a few-shot prompt refers to a prompt that includes multiple examples. A prompt that includes no examples can be referred to as a zero-shot prompt.

[00182] Returning to Figure 9, the model 904 may be formed using any of the training methods and inputs described herein. The model 904 may take inputs from and send outputs to the data obscuration platform (e.g., data obscuration platform 902). For example, the data obscuration platform 902 may send information relating to a third-party platform 906. In some embodiments, the data obscuration platform 902 may send images 908 (e.g., screenshots). For example, the images 908 may relate to the third-party platform 906. In some embodiments, the data obscuration platform 902 may send security information 910. For example, the security information 910 may include security questions, multi-factor authentication

prompts, CAPTCHA tests, or other security hurdles encountered by the data obscuration platform 902. In some embodiments, the data obscuration platform 902 may send errors 912 or other alerts to the model 904. In some embodiments, the data obscuration platform 902 may send other information, such as objectives for data obscuration. In some embodiments, the model 904 may take any of these inputs, or a combination of these inputs, and may generate outputs based on the inputs. For example, the model 904 may generate instructions for satisfying a multi-factor authentication test based. In some embodiments, the model 904 may generate a planner agent based on objectives received by the model 904.

[00183] For example, a planner agent may be generated by a generative AI model (e.g., the model 904), as previously discussed. A generative AI model may be employed to create a planner agent by leveraging its ability to understand and generate complex sequences of tasks based on specified objectives and constraints. The AI model may be trained on a vast dataset of workflows, task sequences, and decision-making processes across various domains, enabling it to learn the intricacies of planning and execution. When tasked with generating a planner agent, the AI model may analyze the given objectives, constraints, and available resources to design a comprehensive workflow. This workflow may include detailed steps, dependencies, and criteria for successful completion. The generative AI model may also incorporate optimization techniques to ensure that the workflow is efficient and effective, minimizing resource usage and execution time. Additionally, the AI model may embed adaptive mechanisms within the planner agent, allowing it to adjust the workflow dynamically based on real-time feedback and changing conditions. By utilizing a generative AI model, the creation of a planner agent becomes a highly automated and intelligent process, capable of producing sophisticated and adaptable workflows tailored to specific objectives and operational environments.

[00184] As discussed above, the set of instructions (e.g., objective-based instructions) may be generated by a generative AI algorithm (e.g., the model 904). For example, a generative AI algorithm may generate the set of instructions based

on inputs including information about the target platform. In some embodiments, the generative AI algorithm may generate the set of instructions based on screenshots of the target platform. In some embodiments, the set of instructions is output by the generative AI algorithm in an instruction-based language. Instruction-based languages for generative AI are designed to provide clear and structured commands that guide the AI in producing specific outputs. Examples of such languages include Python, which is widely used for scripting and automating tasks in AI development due to its simplicity and extensive libraries. Another example is Structured Query Language (SQL), which may be used to instruct AI systems in data retrieval and manipulation tasks within databases. Additionally, JavaScript Object Notation (JSON) is often employed to structure data in a readable format, enabling AI systems to parse and generate outputs based on the provided instructions. These and other instruction-based languages facilitate the interaction between developers and AI systems, ensuring that the generated outputs align with the desired specifications and use cases.

[00185] In some embodiments, a computer vision model (e.g., the model 904) may be used to facilitate generation of sets of instructions. Computer vision models may leverage algorithms and neural networks to interpret and understand visual data. These models may be trained on large datasets of labeled images, allowing them to learn patterns, features, and relationships within the visual content. The training process involves feeding the model numerous examples of images along with their corresponding labels, enabling the model to adjust its internal parameters to minimize errors in its predictions. Once trained, a computer vision model can perform a variety of tasks, such as object detection, image classification, segmentation, and optical character recognition (OCR). For instance, in object detection, the model identifies and locates specific objects within an image, drawing bounding boxes around them. In image classification, the model assigns a label to the entire image based on its content. Segmentation involves dividing an image into regions or segments that correspond to different objects or features. OCR allows the model to extract and interpret text from images.

[00186] A computer vision model may be utilized to take inputs, such as screenshots of website interfaces, and output a set of instructions for navigating the website to change a user's credentials, thereby obscuring the user's data. The process may begin with the model analyzing the screenshots to identify key elements of the website interface, such as buttons, text fields, menus, and links. Using techniques like object detection and OCR, the model may recognize and interpret these elements, understanding their functions and relationships. For instance, the model may identify a "Settings" button, a "Change Password" link, or an "Email Address" text field. Once the elements are identified, the computer vision model may generate a sequence of navigation steps, detailing the actions required to change the user's credentials. These steps may include clicking specific buttons, entering new information into text fields, and confirming changes through dialog boxes. The model may also provide visual annotations on the screenshots, highlighting the elements to interact with and the order of operations. By leveraging computer vision, the model can effectively translate visual inputs into actionable instructions, enabling automated or guided navigation of website interfaces to enhance user privacy and data security. In some embodiments, the data obscuration platform may utilize these and other models alone or in combination with each other to carry out the methods described herein.

[00187] Figure 10 includes a flow diagram of a process 1000 for obscuring communication identifiers associated with users.

[00188] Initially, a data obscuration platform may receive a first request to obscure a first communication identifier associated with a user on a target platform (step 1001). For example, the target platform may be a website, domain, application, or other location. The first communication identifier may be a piece of user data to be obscured. For example, the first communication identifier may be a phone number, email address, or other identifier. Thereafter, the data obscuration platform may retrieve a second communication identifier from a database (step 1002). In some embodiments, the database may include a plurality of communication identifiers. The second communication identifier is different from

the first communication identifier. For example, the second communication identifier may be a different phone number, email address, or other identifier that does not belong to the user.

[00189] The data obscuration platform assigns the second communication identifier to the user for the target platform (step 1003). In some embodiments, the second communication identifier replaces the first communication identifier such that the first communication identifier is no longer accessible via the third-party platform. In this way, the data obscuration platform may obscure the first communication identifier in order to protect the user's data.

[00190] The data obscuration platform receives a second request to direct a communication via the second communication identifier (step 1004). For example, a third party may attempt to transmit a message, call, email, or other communication via the second communication identifier. Since the second communication identifier is not associated with the user, the data obscuration platform must determine how to route the communication to the user. In some embodiments, the data obscuration platform determines that the second request is associated with the target platform (step 1005). For example, the data obscuration platform may determine that the communication originated from the third-party platform or an identifier associated with the third-party platform.

[00191] The data obscuration platform may then direct the communication to the first communication identifier associated with the user via the second communication identifier (step 1006). For example, the data obscuration platform may route the communication to the user at the user's actual contact (e.g., actual phone number) via the replacement contact (e.g., replacement phone number).

[00192] Figure 11 includes a flow diagram of a process 1100 for obscuring credentials associated with users.

[00193] Initially, the data obscuration platform retrieves a user profile that includes credentials associated with a user (step 1101). The credentials may

include passwords, usernames, email addresses, phone numbers, home addresses, work addresses, security questions and answers, payment information, or other data associated with the user. The data obscuration platform may retrieve a set of instructions for obscuring the credentials associated with the user on a target platform (step 1102). In some embodiments, the data obscuration platform may generate the set of instructions, for example, using generative AI. This may enable the data obscuration platform to create instructions for navigating new or updated platforms.

[00194] The data obscuration platform may perform a login on behalf of the user on the target platform using at least one of the credentials associated with the user (step 1103). For example, the data obscuration platform may log into the third-party platform using the user's actual credentials. The data obscuration platform may then generate different credentials for the user (step 1104). For example, the different credentials may include a different password, username, email address, phone number, home address, work address, security question and answer, payment information, or other data that is not associated with the user.

[00195] The data obscuration platform may submit the different credentials to the target platform (step 1105). For example, the data obscuration platform may replace the user's actual credentials with the different credentials. In some embodiments, the data obscuration platform may remove the user's actual credentials from the third-party platform entirely. The data obscuration platform may perform the login on behalf of the user on the target platform using the different credentials associated with the user (step 1106). The data obscuration platform may thereby verify that the different credentials have been accepted by the target platform.

[00196] Finally, the data obscuration platform may update the user profile to include the different credentials (step 1107). For example, the data obscuration platform may save the different credentials in connection with the user profile for future logins.

[00197] Unless contrary to physical possibility, it is envisioned that the steps described above may be performed in various sequences and combinations. For example, the data obscuration platform may obscure identifiers associated with the user based on a request, periodically, in response to a data breach, or based on some other event. As another example, the data obscuration platform may generate different credentials for the user before performing a login, after performing the login, or in parallel with logging in.

Processing System

[00198] Figure 12 is a block diagram illustrating an example of a processing system 1200 that can perform at least some operations described herein. For example, some components of the processing system 1200 may be hosted on a computing device that includes a management platform (e.g., data obscuration platform 102 of Figure 1 or data obscuration platform 212 of Figure 2).

[00199] The processing system 1200 may include a processor 1202, main memory 1206, non-volatile memory 1210, network adapter 1212, video display 1218, input/output device 1220, control device 1222 (e.g., a keyboard or pointing device), drive unit 1224 including a storage medium 1226, and signal generation device 1230 that are communicatively connected to a bus 1216. The bus 1216 is illustrated as an abstraction that represents one or more physical buses or point-to-point connections that are connected by appropriate bridges, adapters, or controllers. The bus 1216, therefore, can include a system bus, a Peripheral Component Interconnect ("PCI") bus or PCI-Express bus, a HyperTransport or industry standard architecture ("ISA") bus, a small computer system interface ("SCSI") bus, a universal serial bus ("USB"), inter-integrated circuit ("I²C") bus, or an IEEE standard 1394 bus (also referred to as "Firewire").

[00200] While the main memory 1206, non-volatile memory 1210, and storage medium 1226 are shown to be a single medium, the terms "machine-readable medium" and "storage medium" should be taken to include a single medium or multiple media (e.g., a centralized/distributed database and/or associated caches

and servers) that store one or more sets of instructions 1228. The terms “machine-readable medium” and “storage medium” shall also be taken to include any medium that is capable of storing, encoding, or carrying a set of instructions for execution by the processing system 1200.

[00201] In general, the routines executed to implement the embodiments of the disclosure may be implemented as part of an operating system or a specific application, component, program, object, module, or sequence of instructions (collectively referred to as “computer programs”). The computer programs typically comprise one or more instructions (e.g., instructions 1204, 1208, 1228) set at various times in various memory and storage devices in a computing device. When read and executed by the processor 1202, the instruction(s) cause the processing system 1200 to perform operations to execute elements involving the various aspects of the present disclosure.

[00202] Further examples of machine- and computer-readable media include recordable-type media, such as volatile memory devices and non-volatile memory devices 1210, removable disks, hard disk drives, and optical disks (e.g., Compact Disk Read-Only Memory (CD-ROMS) and Digital Versatile Disks (DVDs)), and transmission-type media, such as digital and analog communication links.

[00203] The network adapter 1212 enables the processing system 1200 to mediate data in a network 1214 with an entity that is external to the processing system 1200 through any communication protocol supported by the processing system 1200 and the external entity. The network adapter 1212 can include a network adaptor card, a wireless network interface card, a router, an access point, a wireless router, a switch, a multilayer switch, a protocol converter, a gateway, a bridge, bridge router, a hub, a digital media receiver, a repeater, or any combination thereof.

Remarks

[00204] The foregoing description of various embodiments of the claimed

subject matter has been provided for the purposes of illustration and description. It is not intended to be exhaustive or to limit the claimed subject matter to the precise forms disclosed. Many modifications and variations will be apparent to one skilled in the art. Embodiments were chosen and described in order to best describe the principles of the invention and its practical applications, thereby enabling those skilled in the relevant art to understand the claimed subject matter, the various embodiments, and the various modifications that are suited to the particular uses contemplated.

[00205] Although the Detailed Description describes certain embodiments and the best mode contemplated, the technology can be practiced in many ways no matter how detailed the Detailed Description appears. Embodiments may vary considerably in their implementation details, while still being encompassed by the specification. Particular terminology used when describing certain features or aspects of various embodiments should not be taken to imply that the terminology is being redefined herein to be restricted to any specific characteristics, features, or aspects of the technology with which that terminology is associated. In general, the terms used in the following claims should not be construed to limit the technology to the specific embodiments disclosed in the specification, unless those terms are explicitly defined herein. Accordingly, the actual scope of the technology encompasses not only the disclosed embodiments, but also all equivalent ways of practicing or implementing the embodiments.

[00206] The language used in the specification has been principally selected for readability and instructional purposes. It may not have been selected to delineate or circumscribe the subject matter. It is therefore intended that the scope of the technology be limited not by this Detailed Description, but rather by any claims that issue on an application based hereon. Accordingly, the disclosure of various embodiments is intended to be illustrative, but not limiting, of the scope of the technology as set forth in the following claims.

CLAIMS

What is claimed is:

1. A method performed by a computer program executing on a computing device, the method comprising:
 - receiving first input that is representative of a first request to obscure a first communication identifier that is associated with a user of a plurality of users on a target platform;
 - retrieving a second communication identifier from a database that includes a plurality of communication identifiers;
 - assigning the second communication identifier to the user in association with the target platform,
 - wherein the second communication identifier is also assigned to one or more other users of the plurality of users;
 - receiving, from a third party, second input that is representative of a second request to direct a communication via the second communication identifier;
 - determining that the second request is associated with the target platform;
 - and
 - directing the communication to the first communication identifier that is associated with the user via the second communication identifier without revealing the first communication identifier to the third party.
2. The method of claim 1, further comprising:
 - removing the first communication identifier that is associated with the user from the target platform.
3. The method of claim 1, further comprising:
 - assigning, to the user, an extension for the second communication identifier in association with the target platform;

receiving, from the third party, third input that is representative of a third request to direct a second communication via the second communication identifier using the extension; and directing the second communication to the first communication identifier that is associated with the user via the second communication identifier using the extension, without revealing the first communication identifier to the third party.

4. The method of claim 1, further comprising:

receiving, from the third party, third input that is representative of a third request to direct a second communication via the second communication identifier;

determining that the third request is associated with the target platform;

determining that the second communication identifier is assigned, in association with the target platform, to a set of users, the set of users including the user and the one or more other users;

analyzing contents of the second communication to determine whether the contents include a parameter that is associated with the user or whether the parameter that is associated with the user is derivable from the contents or from a source communication identifier from which the second communication originates; and

in response to a determination that the contents include the parameter or that the parameter is derivable from the contents or from the source communication identifier,

directing the second communication to the first communication identifier that is associated with the user via the second communication identifier without revealing the first communication identifier to the third party.

5. The method of claim 4, further comprising:

in response to a determination that the contents do not include the parameter and that the parameter is not derivable from the contents or from the source communication identifier, inputting the contents into a machine learning model to cause the machine learning model to output a prediction as to whether the second communication is intended to be directed to the user; and

in response to a determination that the prediction is that the second communication is intended to be directed to the user,

directing the second communication to the first communication identifier that is associated with the user via the second communication identifier without revealing the first communication identifier to the third party.

6. A method performed by a network-accessible platform that obscures identifiers of users through dynamic assignment of a plurality of communication identifiers, the method comprising:

receiving first input that is representative of a first request to facilitate communication between a first user and a second user of a plurality of users of the network-accessible platform,

wherein the first user is associated with a first communication identifier, and

wherein the second user is associated with a second communication identifier;

retrieving a third communication identifier from a database that includes a plurality of communication identifiers that are dynamically assignable to, and reclaimable from, the plurality of users; and

assigning the third communication identifier to the first and second users such that:

communications that originate from the first communication identifier and are directed to the third communication identifier are routed to the second communication identifier, without

revealing the second communication identifier to the first user,
and

communications that originate from the second communication
identifier and are directed to the third communication identifier
are routed to the first communication identifier, without
revealing the first communication identifier to the second user.

7. The method of claim 6, further comprising:
receiving, from the first user, second input that is representative of a second
request to direct a communication via the third communication
identifier to the second user; and
directing the communication to the second communication identifier of the
second user via the third communication identifier without revealing
the first communication identifier of the first user to the second user
and without revealing the second communication identifier of the
second user to the first user.
8. The method of claim 6, further comprising:
receiving, from a third user, third input that is representative of a third
request to direct a communication to the first user or the second user
using the third communication identifier;
assigning, to the third user, a child communication identifier associated with
the third communication identifier; and
directing the communication to the first communication identifier of the first
user using the child communication identifier without revealing the
first communication identifier of the first user to the third user.
9. The method of claim 6, wherein the first communication identifier includes
a first phone number and the second communication identifier includes a second
phone number, and wherein the third communication identifier includes a
randomized phone number.

10. The method of claim 6, wherein the first communication identifier includes a first email address and the second communication identifier includes a second email address, and wherein the third communication identifier includes a randomized email address.

11. A server computer comprising:
one or more memories storing instructions of a service that is configured to facilitate actions using obscured identifiers associated with users of a platform,
wherein each of the users is associated with a corresponding one of multiple computing devices capable of communicatively coupling to the server computer; and
one or more processors configured to execute the instructions stored on the one or more memories, causing each of the multiple computing devices to:
receive first input that is representative of a first request to obscure an identifier associated with a user of a plurality of users of the platform;
retrieve a different identifier from a database that includes a plurality of different identifiers;
assign the different identifier to the user in association with the platform,
wherein the different identifier is also assigned to one or more other users of the plurality of users;
receive, from a third party, second input that is representative of a second request to perform an action with the user using the different identifier;
determine that the second request is associated with the platform;
and

facilitate the action between the third party and the user using the different identifier without revealing the identifier of the user to the third party.

12. A method performed by a computer program executing on a computing device, the method comprising:

- retrieving a profile that includes one or more credentials associated with a user;

- retrieving a set of instructions for obscuring the one or more credentials associated with the user on a target platform;

- based on the set of instructions:

- performing a login on behalf of the user on the target platform using at least one of the one or more credentials associated with the user;

- generating one or more different credentials for the user;

- submitting the one or more different credentials to the target platform;

- and

- performing the login on behalf of the user on the target platform using the one or more different credentials associated with the user;

- and

- updating the profile to include the one or more different credentials.

13. The method of claim 12, wherein retrieving the profile further comprises extracting the one or more credentials from a plurality of credentials included in the profile, wherein the one or more credentials are to be obscured.

14. The method of claim 12, further comprising submitting, to the target platform, a removal request for removal of the one or more credentials associated with the user from the target platform.

15. The method of claim 12, wherein the set of instructions is generated by a generative artificial intelligence (AI) algorithm based on inputs including information about the target platform.

16. The method of claim 15, wherein the set of instructions is output by the generative AI algorithm in an instruction-based language.

17. The method of claim 12, wherein performing the login on behalf of the user further comprises:

- identifying a prompt, by the target platform, for an authentication code transmitted to a separate account associated with the user;
- retrieving the authentication code from the separate account associated with the user; and
- submitting the authentication code to the target platform.

18. A method comprising:

- retrieving a profile that includes one or more identifiers associated with a user;
- retrieving a set of instructions for obscuring the one or more identifiers associated with the user on a target platform;
- based on the set of instructions:
 - performing a login on behalf of the user on the target platform using at least one of the one or more identifiers associated with the user;
 - generating one or more different identifiers for the user;
 - submitting the one or more different identifiers to the target platform;
 - and
 - performing the login on behalf of the user on the target platform using the one or more different identifiers associated with the user;
 - and
- updating the profile to include the one or more different identifiers.

19. The method of claim 18, further comprising submitting, to the target platform, a removal request for removal of the one or more identifiers associated with the user from the target platform.

20. The method of claim 18, further comprising:
submitting, to the target platform, a removal request for removal of the one or more identifiers associated with the user from the target platform.

21. The method of claim 18, wherein the set of instructions is generated by a generative artificial intelligence (AI) algorithm based on inputs including information about the target platform.

22. The method of claim 18, wherein performing the login on behalf of the user further comprises:

identifying a prompt, by the target platform, for an authentication code transmitted to a separate account associated with the user;
retrieving the authentication code from the separate account associated with the user; and
submitting the authentication code to the target platform.

23. A server computer comprising:
one or more memories storing instructions of a service that is configured to facilitate obscuration of identifiers associated with multiple users, wherein each of the multiple users is associated with a corresponding one of multiple computing devices capable of communicatively coupling to the server computer; and
one or more processors configured to execute the instructions stored on the one or more memories, causing each of the multiple computing devices to:

retrieve a profile that includes one or more identifiers associated with a corresponding user;

retrieve a set of instructions for obscuring the one or more identifiers associated with the corresponding user on a target platform;

based on the set of instructions:

- perform a login on behalf of the corresponding user on the target platform using the one or more identifiers associated with the corresponding user;
- generate one or more different identifiers for the corresponding user;
- submit the one or more different identifiers to the target platform; and
- perform the login on behalf of the corresponding user on the target platform using the one or more different identifiers associated with the corresponding user; and

update the profile to include the one or more different identifiers.