From Zoos to Safaris—From Closed-World Enforcement to Open-World Assessment of Privacy

Michael Backes[1,2], Pascal Berrang[1], and Praveen Manoharan[1(✉)]

[1] Saarland Informatics Campus, CISPA, Saarland University, Saarbrücken, Germany
{backes,berrang,manoharan}@cs.uni-saarland.de
[2] Saarland Informatics Campus, MPI-SWS, Saarbrücken, Germany

Abstract. In this paper, we develop a *user-centric privacy framework for quantitatively assessing the exposure* of personal information in open settings. Our formalization addresses key-challenges posed by such open settings, such as the necessity of user- and context-dependent privacy requirements. As a sanity check, we show that hard non-disclosure guarantees are impossible to achieve in open settings.

In the second part, we provide an instantiation of our framework to address the *identity disclosure* problem, leading to the novel notion of d-convergence to assess the linkability of identities across online communities. Since user-generated text content plays a major role in linking identities between Online Social Networks, we further extend this linkability model to assess the effectiveness of countermeasures against linking authors of text content by their writing style.

We experimentally evaluate both of these instantiations by applying them to suitable data sets: we provide a *large-scale evaluation* of the linkability model on a collection of 15 million comments collected from the Online Social Network Reddit, and evaluate the effectiveness of four semantics-retaining countermeasures and their combinations on the *Extended-Brennan-Greenstadt Adversarial Corpus*. Through these evaluations we validate the notion of d-convergence for assessing the linkability of entities in our Reddit data set and explore the practical impact of countermeasures on the importance of standard writing style features on identifying authors.

1 Introduction

The Internet has undergone dramatic changes in the last two decades, evolving from a mere communication network to a global multimedia platform in which billions of users not only actively exchange information, but increasingly conduct sizable parts of their daily lives. While this transformation has brought tremendous benefits to society, it has also created new threats to online privacy that existing technology is failing to keep pace with. Users tend to reveal personal information without considering the widespread, easy accessibility, potential linkage and permanent nature of online data. Many cases reported in the

A. Aldini et al. (Eds.): FOSAD VIII, LNCS 9808, pp. 87–138, 2016.
DOI: 10.1007/978-3-319-43005-8_3

press show the resulting risks, which range from public embarrassment and loss of prospective opportunities (e.g., when applying for jobs or insurance), to personal safety and property risks (e.g., when sexual offenders or burglars learn users' whereabouts online). The resulting privacy awareness and privacy concerns of Internet users have been further amplified by the advent of the Big-Data paradigm and the aligned business models of personalized tracking and monetizing personal information in an unprecedented manner.

Developing a suitable methodology to reason about the privacy of users in such a large-scale, open web setting, as well as corresponding tool support in the next step, requires at its core a formal privacy model that lives up to the now increasingly dynamic dissemination of unstructured, heterogeneous user content on the Internet: While users traditionally shared information mostly using public profiles with static information about themselves, nowadays they disseminate personal information in an unstructured, highly dynamic manner, through content they create and share (such as blog entries, user comments, a "Like" on Facebook), or through the people they befriend or follow. Furthermore, ubiquitously available background knowledge about a dedicated user needs to be appropriately reflected within the model and its reasoning tasks, as it can decrease a user's privacy by inferring further sensitive information. As an example, Machine Learning and other Information Retrieval techniques provide comprehensive approaches for profiling a user's actions across multiple Online Social Networks, up to a unique identification of a given user's profiles for each such network.

Prior research on privacy has traditionally focused on closed database settings – characterized by a complete view on structured data and a clear distinction of key- and sensitive attributes – and has aimed for strong privacy guarantees using global data sanitization. These approaches, however, are inherently inadequate if such closed settings are replaced by open settings as described above, where unstructured and heterogeneous data is being disseminated, where individuals have a partial view of the available information, and where global data sanitization is impossible and hence strong guarantees have to be replaced by probabilistic privacy assessments.

As of now, *even the basic methodology is missing* for offering users technical means to comprehensively assess the privacy risks incurred by their data dissemination, and their daily online activities in general. Existing privacy models such as k-anonymity [54], l-diversity [40], t-closeness [39] and the currently most popular notion of Differential Privacy [22] follow a database-centric approach that is inadequate to meet the requirements outlined above. We refer the reader to Sect. 3.3 for further discussions on existing privacy models.

1.1 Contribution

In this paper, we present a rigorous methodology for quantitatively assessing the exposure of personal information in open settings. Concretely, the paper makes the following three tangible contributions: (1) a formal framework for reasoning about the disclosure of personal information in open settings, (2) an instantiation

of the framework for reasoning about the identity disclosure problem, and (3) an evaluation of the framework on a collection of 15 million comments collected from the Online Social Network Reddit.

A Formal Framework for Privacy in Open Settings. We propose a novel framework for addressing the essential challenges of privacy in open settings, such as providing a data model that is suited for dealing with unstructured dissemination of heterogeneous information through various different sources and a flexible definition of user-specific privacy requirements that allow for the specification of context-dependent privacy goals. In contrast to most existing approaches, our framework strives to assess the degree of exposure individuals face, in contrast to trying to enforce an individual's privacy requirements. Moreover, our framework technically does not differentiate between non-sensitive and sensitive attributes a-priori, but rather starts from the assumption that all data is equally important and can lead to privacy risks. More specifically, our model captures the fact that the sensitivity of attributes is highly user- and context-dependent by deriving information sensitivity from each user's privacy requirements. As a sanity check we prove that hard non-disclosure guarantees cannot be provided for the open setting in general, providing incentive for novel approaches for assessing privacy risks in the open settings.

Reasoning about Identity Disclosure in Open Settings. We then instantiate our general privacy framework for the specific use case of identity disclosure. Our framework defines and assesses identity disclosure (i.e., identifiability and linkability of identities) by utilizing entity similarity, i.e., an entity is private in a collection of entities if it is sufficiently similar to its peers. At the technical core of our model is the new notion of d-convergence, which quantifies the similarity of entities within a larger group of entities. It hence provides the formal grounds to assess the ability of any single entity to blend into the crowd, i.e., to hide amongst peers. The d-convergence model is furthermore capable of assessing identity disclosure risks specifically for single entities. To this end, we extend the notion of d-convergence to the novel notion of (k, d)-anonymity, which allows for entity-centric identity disclosure risk assessments by requiring d-convergence in the local neighborhood of a given entity. Intuitively, this new notion provides a generalization of k-anonymity that is not bound to matching identities based on pre-defined key-identifiers.

Empirical Evaluation on Reddit. Third, we perform an instantiation of our identity disclosure model for the important use case of analyzing user-generated text content in order to characterize specific user profiles. We use unigram frequencies extracted from user-generated content as user attributes, and we subsequently demonstrate that the resulting unigram model can indeed be used for quantifying the degree of anonymity of – and ultimately, for differentiating – individual entities. For the sake of exposition, we apply this unigram model to a collection of 15 million comments collected from the Online Social Network Reddit. The computations were performed on two Dell PowerEdge R820 with 64 virtual

cores each at 2.60 GHz over the course of six weeks. Our evaluation shows that (k, d)-anonymity suitably assesses an identity's anonymity and provides deeper insights into the data set's structure.

Assessing the Effectiveness of Countermeasures Against Authorship Recognition. Fourth, by extending the linkability model model introduced in the second step, we develop a novel measure for assessing the importance of stylometric features for the identifiability of authors. We adapt and extend the user models introduced in the general framework to fit our use case of authorship recognition, effectively defining a model for writing style that allows us to capture a comprehensive list of stylometric features, as introduced by Abbasi and Chen [3]. Overall, we develop a model of the authorship recognition problem that allows us to formally reason about authorship recognition in the open setting of the Internet.

By using these writing-style models, we then derive how we can identify important stylometric features that significantly contribute to the identification of the correct author from the context in which text is published. We employ standard regression and classification techniques to determine the importance of each type of stylometric feature. From this importance assessment we then further derive the *gain* measure for the effectiveness of countermeasures against authorship identification by measuring how well they reduce the importance of stylometric features.

Countermeasure Evaluation. Finally, we apply this measure to assess the effectiveness of four automatic countermeasures, namely synonym substitution, spell checking, special character modification and adding/removing misspellings. In this evaluation, we follow a general and comprehensive methodology that structures the evaluation process and is easily extensible for future evaluation.

We perform our experiments on the Extended-Brennan-Greenstadt Adversarial Corpus consisting of texts written by 45 different authors. Each author contributed at least 6500 words to the corpus [11].

1.2 Outline

We begin by discussing related work in Sect. 2 and explain why existing privacy notions are inadequate for reasoning about privacy in open web settings in Sect. 3. We then define our privacy framework in Sect. 4 and instantiate it for reasoning about identity disclosure in Sect. 5. In Sect. 6 we perform a basic evaluation of the identity disclosure model on the Reddit Online Social Network. We extend the identity disclosure model to a model for assessing the effectiveness of countermeasures against authorship recognition in Sect. 7, which we then also evaluate on Reddit in Sect. 8. We summarize our findings Sect. 9.

2 Related Work

In this section, we give an overview over other relevant related work that has not yet been considered in the previous subsection.

Privacy in Closed-World Settings. The notion of privacy has been exhaustively discussed for specific settings such as statistical databases, as well as for more general settings. Since we already discussed the notions of k-anonymity [54], l-diversity [40] t-closeness [39] and Differential Privacy [22] in Sect. 3.3 in great detail, we will now discuss further such notions.

A major point of criticism of Differential Privacy, but also the other existing privacy notions, found in the literature [9,35] is the (often unclear) trade-off between utility and privacy that is incurred by applying database sanitation techniques to achieve privacy. Several works have shown that protection against attribute disclosure cannot be provided in settings that consider an adversary with arbitrary auxiliary information [21,23,24]. We later show, as sanity check, that in our formalization of privacy in open settings, general non-disclosure guarantees are indeed impossible to achieve. By providing the necessary formal groundwork in this paper, we hope to stimulate research on *assessing* privacy risks in open settings, against explicitly spelled-out adversary models.

Kasiviswanathan and Smith [34] define the notion of ϵ-semantic privacy to capture general non-disclosure guarantees. We define our adversary model in a similar fashion as in their formalization and we use ϵ-semantic privacy to show that general non-disclosure guarantees cannot be meaningfully provided in open settings.

Several extensions of the above privacy notions have been proposed in the literature to provide privacy guarantees in use cases that differ from traditional database privacy [7,15,16,30,59,61]. These works aim at suitably transforming different settings into a database-like setting that can be analyzed using differential privacy. Such a transformation, however, often abstracts away from essential components of these settings, and as a result achieve impractical privacy guarantees. As explained in Sect. 3.3, the open web setting is particularly ill-suited for such transformations.

Specifically for the use case in Online Social Networks (in short, OSNs), many works [16,30,37,59,61] apply the existing database privacy notions for reasoning about attribute disclosure in OSN data. These works generally impose a specific structure on OSN data, such as a social link graph, and reason about the disclosure of private attributes through this structure. Zhaleva et al. [59] show that mixed public and private profiles do not necessarily protect the private part of a profile since they can be inferred from the public part. Heatherly et al. [30] show how machine learning techniques can be used to infer private information from publicly available information. Kosinksi et al. [37] moreover show that machine learning techniques can indeed be used to predict personality traits of users and their online behavior. Zhou et al. [61] apply the notions of k-anonymity and l-diversity to data protection in OSNs and discuss the complexity of finding private subsets. Their approach does however suffer from the same problems these techniques have in traditional statistical data disclosure, where an adversary with auxiliary information can easily infer information about any specific user. Chen et al. [16] provide a variation of differential privacy which allows for privacy and protection against edge-disclosure attacks in the correlated

setting of OSNs. The setting, however, remains static, and it is assumed that the data can be globally sanitized in order to provide protection against attribute disclosure. Again, as discussed in Sect. 3.3, this does not apply to the open web setting with its highly unstructured dissemination of data.

Privacy in Online Social Networks. A growing body of research shows that commonly used machine learning and information retrieval techniques can be used to match a user's profiles across different OSNs [13,19] or to identify the unique profile of a given user [8,17,53]. Scerri et al., in particular, present the digital.me framework [51,52] which attempts to unify a user's social sphere across different OSNs by, e.g., matching the profiles of the same user across these OSNs. While their approach is limited to the closed environment they consider, their work provides interesting insights into identity disclosure in more open settings.

Several works in the literature (e.g., [38,41]) have focused on the protection of so-called Personally Identifiable Information (PII) introduced in privacy and data-protection legislation [2], which constitute a fixed set of entity attributes that even in isolation supposedly lead to the unique identification of entities. Narayanan and Shmatikov, however, show that the differentiation between key attributes that identify entities, and sensitive attributes that need to be protected, is not appropriate for privacy in pervasive online settings such as the Internet [47,48]. Technical methods for identifying and matching entities do not rely on the socially perceived sensitivity of attributes for matching, but rather any combination of attributes can lead to successful correlation of corresponding profiles. Our privacy model treats every type of entity attribute as equally important for privacy and allows for the identification of context-dependent, sensitive attributes.

Authorship Recognition. The field of linguistic stylometry is a is a widely explored topic in the literature [3,36,43,57]. This starts from pre-computer approaches to identifying text-authors based on simple text features such as word-length [43] to the, nowadays, machine-learning centered approaches that try to include a plethora of statistical features to correctly identify the author of a given text [3,36,57].

Stylometry has successfully been utilized in various areas: as an assisting tool in historical research [31,49], allowing for the correct attribution of text with previously unknown origin, or providing evidence in criminal investigations [12,14].

With the rise of the Internet as a large-scale communication platform for end-users, however, stylometry now also poses a significant threat to user privacy. As shown by Narayanan *et al.* [46], it is entirely feasible to identify the authors of, e.g., blog-posts on a scale as large as the Internet. Afroz *et al.* [5] also show that authors of private messages in underground forums can effectively be de-anonymized by stylometry.

Adversarial Stylometry. Several works have shown that hiding an author's identity is indeed possible by means of obfuscation and imitation [10,42].

In particular, Brennan *et al.* [10] show that, for text corpora with at least 6500 words per author, applying methods such as asking the authors to rewrite their texts or doubly translating with machine translation can indeed reduce the accuracy of state-of-the-art stylometric methods. They also provide an implementation of their ideas in Anonymouth [6], a semi-automatic tool, assisting users in anonymizing their writing style by identifying critical text features and asking them to rewrite corresponding text passages. This work, however, only provides results for text corpora with large amounts of text per author and is based on the same dataset as ours.

Authorship obfuscation can also be detected, as shown by various work in the literature [4,33,50]. However, these works again require text corpora with large amounts of text per author. It would be interesting to see the effectiveness of these obfuscation-detection methods in the online setting with much less text per author.

3 Privacy in Open Settings

Before we delve into the technical parts of this paper, we give an informal overview over privacy in the Internet of the future. To this end, we first provide an example that illustrates some of the aspects of privacy in the Internet, and then in detail discuss the challenges of privacy in the Internet and why existing privacy notions are not applicable to this setting.

3.1 Example

Consider the following example: Employer Alice receives an application by potential employee Bob which contains personal information about Bob. Before she makes the decision on the employment of Bob, however, she searches the internet and tries to learn even more about her potential employee. A prime source of information are, for example, Online Social Networks (OSNs) which Alice can browse through. If she manages to identify Bob's profile in such an OSN she can then learn more about Bob by examining the publicly available information of this profile.

In order to correctly identify Bob's profile in an OSN, Alice takes the following approach: based on the information found in Bob's application, she constructs a model θ_B that contains all attributes, such as name, education or job history, extracted from Bob's application. She then compares this model θ_B to the profiles $P_1, \ldots, P_n$ found in the OSNs and ranks them by similarity to the model θ_B. Profiles that show sufficient similarity to the model θ_B are then chosen by Alice as belonging to Bob. After identifying the (for Alice) correctly matching profiles $P_1^*, \ldots, P_i^*$ of Bob, Alice can finally merge their models $\theta_1^*, \ldots, \theta_i^*$ with θ_B to increase her knowledge about Bob.

Bob now faces the problem that Alice could learn information about him that he does not want her to learn. He basically has two options: he either does not share this critical information at all, or makes sure that his profile is not

identifiable as his. In OSNs such as Facebook, where users are required to identify themselves, Bob can only use the first option. In anonymous or pseudonymous OSNs such as Reddit or Twitter, however, he can make use of the second option. He then has to make sure that he does not share enough information on his pseudonymous profiles that would allow Alice to link his pseudonymous profile to him personally.

Privacy in the open web is mostly concerned with the second option: we cannot protect an entity ϵ against sharing personal information through a profile which is already uniquely identified with the entity ϵ. We can, however, estimate how well an pseudonymous account of ϵ can be linked to ϵ, and through this link, learn personal information about ϵ. As the example above shows, we can essentially measure privacy in terms of similarity of an entity ϵ in a collection of entities $\mathcal{E}$.

The identifiability of ϵ then substantially depends on the attributes ϵ exhibits in the context of $\mathcal{E}$ and does not necessarily follow the concept of personally identifiable information (PII) as known in the more common understanding of privacy and in privacy and data-protection legislation [2]: here, privacy protection only goes as far as protecting this so-called personally identifiable information, which often is either not exactly defined, or restricted to an a-priori-defined set of attributes such as name, Social Security number, etc. We, along with other authors in the literature [47,48], find however that the set of critical attributes that need to be protected differ from entity to entity, and from community to community. For example, in a community in which all entities have the name "Bob", exposing your name does not expose any information about yourself. In a different community, however, where everyone has a different name, exposing your name exposes a lot of information about yourself.

In terms of the privacy taxonomy formulated by Zheleva and Getoor [60], the problem we face corresponds to the identity disclosure problem, where one tries to identify whether and how an identity is represented in an OSN. We think that this is one of the main concerns of users of frequently used OSNs, in particular those that allow for pseudonymous interactions: users are able to freely express their opinions in these environments, assuming that their opinions cannot be connected to their real identity. However, any piece of information they share in their interactions can leak personal information that can lead to identity disclosure, defeating the purpose of such pseudonymous services.

To successfully reason about the potential disclosure of sensitive information in such open settings, we first have to consider various challenges that have not been considered in traditional privacy research. After presenting these challenges, we discuss the implications of these challenges on some of the existing privacy notions, before we consider other relevant related work in the field.

3.2 Challenges of Privacy in Open Settings

In this subsection, we introduce the challenges induced by talking about privacy in open settings:

(C1) Modeling Heterogeneous Information. We require an information model that allows for modeling various types of information and that reflects the heterogeneous information shared throughout the Internet. This models needs to adequately represent personal information that can be inferred from various sources, such as static profile information or from user-generated content, and should allow statistical assessments about the user, as is usually provided by knowledge inference engines. We propose a solution to this challenge in Sect. 4.1.

(C2) User-Specified Privacy Requirements. We have to be able to formalize user-specified privacy requirements. This formalization should use the previously mentioned information model to be able to cope with heterogeneous information, and specify which information should be protected from being publicly disseminated. We present a formalization of user privacy requirements in Sect. 4.4.

(C3) Information Sensitivity. In open settings, information sensitivity is a function of user expectations and context: we therefore need to provide new definitions for sensitive information that takes user privacy requirements into account. We present context- and user-specific definitions of information sensitivity in Sect. 4.5.

(C4) Adversarial Knowledge Estimation. To adequately reason about disclosure risks in open settings we also require a parameterized adversary model that we can instantiate with various assumptions on the adversary's knowledge: this knowledge should include the information disseminated by the user, as well as background knowledge to infer additional information about the user. In Sect. 4, we define our adversary model based on statistical inference.

In the following sections, we provide a rigorous formalization for these requirements, leading to a formal framework for privacy in open settings. We will instantiate this framework in Sect. 5.3 to reason about the identity disclosure in particular.

We begin by discussing why existing privacy notions are not suited for reasoning about privacy in open settings. Afterwards, we provide an overview over further related work.

3.3 Inadequacy of Existing Models

Common existing privacy notions such as k-anonymity [54], l-diversity [40], t-closeness [39] and the currently most popular notion of Differential Privacy [22] provide the technical means for privacy-friendly data-publishing in a closed-world setting: They target scenarios in which all data is available from the beginning, from a single data source, remains static and is globally sanitized in order to provide rigorous privacy guarantees. In what follows, we describe how these notions fail to adequately address the challenges of privacy in open settings discussed above.

(a) Absence of Structure and Classification of Data. All the aforementioned privacy models require an a-priori structure and classification of the data under consideration. Any information gathered about an individual thus has to be embedded in this structure, or it cannot be seamlessly integrated in these models.

(b) No Differentiation of Attributes. All of these models except for Differential Privacy require an additional differentiation between key attributes that identify an individual record, and sensitive attributes that a users seeks to protect. This again contradicts the absence of an a-priori, static structure in our setting. Moreover, as pointed out above and in the literature [48], such a differentiation cannot be made a-priori in general, and it would be highly context-sensitive in the open web setting.

(c) Ubiquitously Available Background Knowledge. All of these models, except for Differential Privacy, do not take into account adversaries that utilize ubiquitously available background knowledge about a target user to infer additional sensitive information. A common example of background knowledge is openly available statistical information that allows the adversary to infer additional information about an identity.

(d) Privacy for Individual Users. All these models provide privacy for the whole dataset, which clearly implies privacy of every single user. One of the major challenges in open settings such as the Internet, however, is that accessing and sanitizing all available data is impossible. This leads to the requirement to design a local privacy notion that provides a lower privacy bound for every individual user, even if we only have partial access to the available data.

The notion of Differential Privacy only fails to address some of the aforementioned requirements (parts *a* and *d*), but it comes with the additional assumption that the adversary knows almost everything about the data set in question (everything except for the information in one database entry). This assumption enables Differential Privacy to avoid differentiation between key attributes and sensitive attributes. This strong adversarial model, however, implies that privacy guarantees are only achievable if the considered data is globally perturbed [21,23,24], which is not possible in open web settings.

The conceptual reason for the inadequacy of existing models for reasoning about privacy in open web settings is mostly their design goal: Privacy models have thus far mainly been concerned with the problem of attribute disclosure within a single data source: protection against identity disclosure was then attempted by preventing the disclosure of any (sensitive) attributes of a user to the public. In contrast to static settings such as private data publishing, where we can decide which information will be disclosed to the adversary, protection against any attribute disclosure in open settings creates a very different set of challenges which we will address in the following sections.

4 A Framework for Privacy in Open Settings

In this section, we first develop a user model that is suited for dealing with the information dissemination behavior commonly observed on the Internet. We then formalize our adversary model and show, as a sanity check, that hard privacy guarantees cannot be achieved in open settings. We conclude by defining privacy goals in open settings through user-specified privacy requirements from which we then derive a new definition of information sensitivity suited to open settings.

4.1 Modeling Information in Open Settings

We first define the notion of entity models and restricted entity models. These models capture the behavior of these entities and in particular describe which attributes an entity exhibits publicly.

Definition 1 (Entity Model). *Let $\mathcal{A}$ be the set of all attributes. The* entity model θ_ϵ *of an entity ϵ provides for all attributes $\alpha \in \mathcal{A}$ an* attribute value $\theta_\epsilon(\alpha) \in \mathsf{dom}(\alpha) \cup \{\mathsf{NULL}\}$ *where* $\mathsf{dom}(\alpha)$ *is the domain over which the attribute α_i is defined.*

The domain $\mathsf{dom}(\theta)$ *of an entity model θ is the set of all attributes $\alpha \in \mathcal{A}$ with value $\theta(\alpha) \neq \mathsf{NULL}$.*

An entity model thus corresponds to the information an entity can publicly disseminate. With the specific null value NULL we can also capture those cases where the entity does not have any value for that specific attribute.

In case the adversary has access to the full entity model, a set of entity models basically corresponds to a database with each attribute $\alpha \in \mathcal{A}$ as its columns. In the open setting, however, an entity typically does not disseminate all attribute values, but instead only a small part of them. We capture this with the notion of restricted entity models.

Definition 2 (Restricted Entity Model). *The* restricted entity model $\theta_\epsilon^{\mathcal{A}'}$ *is the entity model of ϵrestricted to the non empty attribute set $\mathcal{A}' \neq \emptyset$, i.e.,*

$$\theta_\epsilon^{\mathcal{A}'}(\alpha) = \begin{cases} \theta_\epsilon(\alpha), & \text{if } \alpha \in \mathcal{A}' \\ \mathsf{NULL}, & \text{otherwise} \end{cases}$$

In the online setting, each of the entities above corresponds to an online profile. A user u usually uses more than one online service, each with different profiles $P_1^u, \ldots, P_l^u$. We thus define a user model as the collection of the entity models describing each of these profiles.

Definition 3 (User Model / Profile Model). *The* user model $\theta_u = \{\theta_{P_1^u}, \ldots, \theta_{P_1^u}\}$ *of a user u is a set of the entity models $\theta_{P_1^u}, \ldots, \theta_{P_1^u}$, which we also call* profile models.

With a user model that separates the information disseminated under different profiles, we will be able to formulate privacy requirements for each of these profiles separately. We will investigate this in Sect. 4.4.

4.2 Adversary Model

In the following we formalize the adversary we consider for privacy in open settings. In our formalization, we follow the definitions of a semantic, Bayesian adversary introduced by Kasiviswanathan and Smith [34].

For any profile P, we are interested in what the adversary Adv learns about P observing publicly available information from P. We formalize this learning process through *beliefs* on the models of each profile.

Definition 4 (Belief). *Let $\mathcal{P}$ be the set of all profiles and let $\mathcal{D}_{\mathcal{A}}$ be the set of all distributions over profile models. A* belief $b = \{b_P | P \in \mathcal{P}\}$ *is a set of distributions $b_P \in \mathcal{D}_{\mathcal{A}}$.*

We can now define our privacy adversary in open settings using the notion of belief above.

Definition 5 (Adversary). *An adversary* Adv *is a pair of* prior *belief b and world knowledge κ, i.e.,* $\mathsf{Adv} = (b, \kappa)$.

The adversary Adv's prior belief b represents his belief in each profile's profile model before makes any observations. This prior belief can, in particular, also include background knowledge about each profile P. The world knowledge κ of the adversary represents a set of inference rules that allow him to infer additional attribute values about each profile from his observations.

We next define the publicly observations based on which the adversary learns additional information about each profile.

Definition 6 (Publication Function). *A* publication function *G is a randomized function that maps each profile model θ_P to a restricted profile model $G(\theta_P) = \theta_P^{\mathcal{A}'}$ such that there is at least one attribute $\alpha \in \mathcal{A}'$ with $\theta_P(\alpha) = G(\theta_P)(\alpha)$.*

The publication function G reflects which attributes are disseminated publicly by the user through his profile P. G can, in particular, also include local sanitization where some attribute values are perturbed. However, we do require that at least one attribute value remains correct to capture utility requirements faced in open settings.

A public observation now is the collection of all restricted profile models generated by a publication function.

Definition 7 (Public Observation). *Let $\mathcal{P}$ be the set of all profiles, and let G be a publication function. The* public observation *$\mathcal{O}$ is the set of all restricted profile models generated by G, i.e., $\mathcal{O} = \{G(\theta_P) | P \in \mathcal{P}\}$.*

The public observation $\mathcal{O}$ essentially captures all publicly disseminated attribute values that can be observed by the adversary. Given such an observation $\mathcal{O}$, we can now determine what the adversary Adv learns about each profile by determining his *a-posteriori belief.*

Definition 8 (A-Posteriori Belief). *Let $\mathcal{P}$ be the set of all profiles. Given an adversary* $\mathsf{Adv} = (b, \kappa)$ *and a public observation $\mathcal{O}$, the adversary's* a-posteriori belief $\overline{b} = \{\overline{b}_P \in \mathcal{D}_{\mathcal{A}} | P \in \mathcal{P}\}$ *is determined by applying the Bayesian inference rule, i.e.,*

$$\overline{b}_P[\theta|\mathcal{O}, \kappa] = \frac{Pr[\mathcal{O}|\kappa, \theta] \cdot b_P[\theta]}{\sum_{\theta'} Pr[\mathcal{O}|\kappa, \theta'] \cdot b_P[\theta']}.$$

Here, the conditional probability $Pr[\mathcal{O}|\kappa, \theta]$ describes the likelihood that the observational $\mathcal{O}$ is created by the specific entity model θ.

We will utilize the a-posteriori belief of the adversary to reason about the violation of the user specified privacy requirements in Sect. 4.4.

4.3 Inapplicability of Statistical Privacy Notions

In the following, we formally show that traditional non-disclosure guarantees, e.g., in the style of Differential Privacy, are not possible in open settings.

Kasiviswanathan and Smith [34] provide a general definition of non-disclosure they call ϵ-privacy. In their definition, they compare the adversary Adv's a-posteriori beliefs after observing the transcript t generated from a database sanitazitaion mechanism $\mathcal{F}$ applied on two adjacent databases with n rows: first on the database x, leading to the belief $\overline{b}_0[.|t]$, and secondly on the database x_{-i}, where a value in the ith row in x is replaced by a default value, leading to the belief $\overline{b}_i[.|t]$.

Definition 9 (ϵ-semantic Privacy [34]). *Let $\epsilon \in [0, 1]$. A randomized algorithm $\mathcal{F}$ is ϵ-semantically private if for all belief distributions b on D^n, for all possible transcripts, and for all $i = 1 \ldots n$:*

$$\mathsf{SD}(\overline{b}_0[.|t], \overline{b}_i[.|t]) \leq \epsilon.$$

Here, SD is the total variation distance of two probability distributions.

Definition 10. *Let X and Y be two probability distributions over the sample space D. The* total variation distance SD *of X and Y is*

$$\mathsf{SD}(X, Y) = \mathsf{max}_{S \subset D} \left[Pr[X \in S] - Pr[Y \in S] \right].$$

Kasiviswanathan and Smith [34] show that ϵ-differential privacy is essentially equivalent to ϵ-semantic privacy.

In our formalization of privacy in open settings, varying a single database entry corresponds to changing the value of a single attribute α in the profile model θ_P of a profile P to a default value. We denote this modified entity model with θ_P^{α}, and the thereby produced a-posteriori belief by $\overline{b}_P^{\alpha}$. A profile P would then be ϵ-semantically private if for any modified profile model θ_P^{α}, the a-posteriori belief of adversary Adv does not change by more than ϵ.

Definition 11 (ϵ-semantic Privacy in Open Settings). *Let $\epsilon \in [0,1]$. A profile P is ϵ-semantically private in open settings if for any attribute α,*

$$\mathsf{SD}(\overline{b}_P[.|\mathcal{O}], \overline{b}^{\alpha}_P[.|\mathcal{O}]) \leq \epsilon$$

where $\overline{b}_P$ and $\overline{b}^{\alpha}_P$ are the a-posteriori beliefs of the adversary after observing the public output of θ_P and θ^{α}_P respectively.

As expected, we can show that ϵ-semantic privacy can only hold for $\epsilon = 1$ in open settings.

Theorem 1. *For any profile model θ_P and any attribute α, there is an adversary* Adv *such that*

$$\mathsf{SD}(\overline{b}[.|\mathcal{O}], \overline{b}^{\alpha}[.|\mathcal{O}]) \geq 1.$$

Proof. Let Adv have a uniform prior belief, i.e., all possible profile models have the same probability, and empty world knowledge κ. Let αbe the one attribute that remains the same after applying the publication function G. Let x be the original value of this attribute α and let x^* be the default value that replaces x.

Observing the restricted profile model $\theta_P[\mathcal{A}']$ without any additional world knowledge will lead to an a-posteriori belief, where the probability of the entity model θ with $\theta[\mathcal{A}'] = \theta_P[\mathcal{A}']$ and NULL everywhere else, is set to 1.

Conversely, the modified setting will result in an a-posteriori belief that sets the probability for the entity model θ^* to one, where θ^* is constructed for the modified setting as θ above. Thus $\overline{b}[\theta|\mathcal{O}] = 1$, whereas $\overline{b}^{\alpha}[\theta|\mathcal{O}] = 0$, and hence $\mathsf{SD}(\overline{b}[.|\mathcal{O}], \overline{b}^{\alpha}[.|\mathcal{O}]) = 1$. □

Intuitively, the adversary can easily distinguish differing profile models because (a) he can directly observe the profiles publicly available information, (b) he chooses which attributes he considers for his inference and (c) only restricted, local sanitization is available to the profile. Since these are elementary properties of privacy in open settings, we can conclude that hard security guarantees in the style of differential privacy are impossible to achieve in open settings.

However, we can provide an assessment of the disclosure risks by explicitly fixing the a-priori knowledge and the attribute set considered by the adversary. While we no longer all-quantify over all possible adversaries, and therefore lose the full generality of traditional non-disclosure guarantees, we might still provide meaningful privacy assessments in practice. We further discuss this approach in Sect. 4.5, and follow this approach in our instantiation of the general model for assessing the likelihood of identity disclosure in Sect. 5.

4.4 User-Specified Privacy Requirements

In the following we introduce user-specified privacy requirements that allow us to formulate privacy goals that are user- and context-dependent. These can then lead to restricted privacy assessments instead of general privacy guarantees that we have shown to be impossible in open setting in the previous section.

We define a user's privacy requirements on a per-profile basis, stating which attribute values should not be inferred by adversary after seeing a public observations $\mathcal{O}$.

Definition 12 (Privacy Policy). *A* privacy policy $\mathcal{R}$ *is a set of privacy requirements* $r = (P, \{\alpha_i = x_i\})$ *which require that profile* P *should never expose the attribute values* x_i *for the attributes* $\alpha_i \in \mathcal{A}$.

By setting privacy requirements in a per-profile basis we capture an important property of information dissemination in open settings: users utilize different profiles for different context (e.g., different online services) assuming these profiles remain separate and specific information is only disseminated under specific circumstances.

Given the definition of privacy policies, we now define the violation of a policy by considering the adversary's a-posterior belief $\overline{b}$, as introduced in Sect. 4.2.

Definition 13 (Privacy Policy Satisfaction / Violation). *Let* $\mathsf{Adv} = (b, \kappa)$ *be an adversary with a-posteriori belief* $\overline{b}$*, and let* $\theta[\alpha = x]$ *be the set of all entity models that have the value* x *for the attribute* α*. A profile* P_i^u σ-satisfies *a user's privacy requirement* $r_j^u = (P, \{\alpha_i = x_i\})$*, written* $P_i^u \models_\sigma r_j^u$*, if*

- $P = P_i^u$
- $\forall \alpha_i : \sum_{\theta \in \theta[\alpha_i = x_i]} \overline{b}_P[\theta|\mathcal{O}, \kappa] \leq \sigma$

and σ-violates *the user's privacy requirement otherwise.*

A user model θ_u σ-satisfies *a user* u*'s privacy policy* $\mathcal{R}_u$*, written* $\theta_u \models_\sigma \mathcal{R}_u$*, if all profile models* $\theta_{P_i^u}$ σ*-satisfy their corresponding privacy requirements, and* σ-violates *the privacy policy otherwise.*

The above attributes can also take the form of "P belongs to the same user as P'", effectively restricting which profiles should be linked to each other. We will investigate this profile linkability problem specifically in Sect. 5.

4.5 Sensitive Information

In contrast to the closed-world setting, with its predefined set of sensitive attributes that automatically defines the privacy requirements, a suitable definition of information sensitivity in the open setting is still missing. In the following, we derive the notion of sensitive information from the user privacy requirements we defined in Sect. 4.4.

Definition 14 (Sensitive Attributes). *A set of attributes* $\mathcal{A}^*$ *is* sensitive *for a user* u *in the context of her profile* P_i^u *if* u*'s privacy policy* $\mathcal{R}_u$ *contains a privacy requirement* $r = (P_i^u, \mathcal{A}' = X)$ *where* $\mathcal{A}^* \subseteq \mathcal{A}'$.

Here, we use the notation $\mathcal{A} = X$ as vector representation for $\forall \alpha_i \in \mathcal{A} : \alpha_i = x_i$.

Sensitive attributes, as defined above, are not the only type of attributes that are worth to protect: In practice, an adversary can additionally infer sensitive attributes from other attributes through statistical inference using a-priori knowledge. We call such attributes that allow for the inference of sensitive attributes *critical attributes*.

Definition 15 (Critical Attributes). *Given a set of attributes $\mathcal{A}^*$, let P be a profile with $\mathsf{dom}(\theta_P) \supseteq \mathcal{A}$, and let P' be the profile with the restricted profile model $\theta_{P'} = \theta_P^{\mathcal{A}'}$, where $\mathcal{A}' = \mathsf{dom}(\theta_P) \setminus \mathcal{A}^*$.*

The set of attributes $\mathcal{A}^*$ is σ-critical *for the user u that owns the profile P and an adversary with prior belief b_P and world knowledge κ, if u's privacy policy $\mathcal{R}_u$ contains a privacy requirement r such that P σ-violates r but P' does not.*

Critical information require the same amount of protection as sensitive information, the difference however being that critical information is only protected for the sake of protecting sensitive information.

As a direct consequence of the definition above, sensitive attributes are also critical.

Corollary 1. *Let $\mathcal{A}$ be a set of sensitive attributes. Then $\mathcal{A}$ is also* 0*-critical.*

Another consequence we can draw is that privacy requirements will always be satisfied if no critical attributes are disseminated.

Corollary 2. *Let $\mathcal{O}$ be a public observations that does not include any critical attributes for a user u and an adversary* Adv*. Then u's privacy policy $\mathcal{R}_u$ is σ-satisfied against* Adv*.*

The corollary above implies that, while we cannot provide general non-disclosure guarantees in open settings, we can provide privacy assessments for specific privacy requirements, given an accurate estimate of the adversary's prior beliefs.

While privacy assessments alone are not satisfactory from a computer security perspective, where we usually require hard security guarantees quantified over all possible adversaries, the fact remains that we are faced with privacy issues in open settings that are to this day unanswered for due to the impossibility of hard guarantees in such settings. Pragmatically thinking, we are convinced that we should move from impossible hard guarantees to more practical privacy assessments instead. This makes particularly sense in settings where users are not victims of targeted attacks, but instead fear attribute disclosure to data-collecting third parties.

5 Linkability in Open Settings

In the following we instantiate the general privacy model introduced in the last section to reason about the likelihood that two profiles of the same user are linked by the adversary in open settings. We introduce the novel notion of (k, d)-anonymity with which we assess anonymity and linkability based on the similarity of profiles within an online community.

To simplify the notation we introduce in this section, we will, in the following, talk about matching *entities* ϵ and ϵ' the adversary wants to link, instead of profiles P_1 and P_2 that belong to the same user u. All definitions introduced in the general framework above naturally carry over to entities as well.

5.1 Model Instantiation for Linkability

In the linkability problem, we are interested in assessing the likelihood that two matching entities ϵ and ϵ' can be linked, potentially across different online platforms. The corresponding privacy requirements, as introduced in Sect. 4.4, are $r_1 = (\epsilon, \alpha_L)$ and $r_2 = (\epsilon', \alpha_L)$, where α_L is the attribute that ϵ and ϵ' belong to the same user. Consequently, we say that these entities are unlinkable if they satisfy the aforementioned privacy requirements.

Definition 16 (Unlinkability). *Two entities ϵ and ϵ' are σ-unlinkable if $\{\theta_\epsilon, \theta_{\epsilon'}\} \models_\sigma \{r_1, r_2\}$.*

5.2 Anonymity

To assess the identity disclosure risk of an entity ϵ within a collection of entities $\mathcal{E}$, we use the following intuition: ϵ is anonymous in $\mathcal{E}$ if there is a subset $\mathcal{E}' \subseteq \mathcal{E}$ to which ϵ is very similar. The collection $\mathcal{E}$' then is an anonymous subset of $\mathcal{E}$ for ϵ.

To assess the similarity of entities within a collection of entities, we will use a distance measure dist on the entity models of these entities. We will require that this measure provides all properties of a metric.

A collection of entities in which the distance of all entities to ϵ is small (i.e., $\leq$ a constant d) is called d-convergent for ϵ.

Definition 17. *A* collection *of entities $\mathcal{E}$ is d-convergent for ϵ if $\mathsf{dist}(\theta_\epsilon, \theta_{\epsilon'}) \leq d$ for all $\epsilon' \in \mathcal{E}$.*

Convergence measures the similarity of a collection of individuals. Anonymity is achieved if an entity can find a collection of entities that are all similar to this entity. This leads us to the definition of (k, d)-anonymity, which requires a subset of similar entities of size k.

Definition 18. *An entity ϵ is* (k, d)-anonymous *in a collection of entities $\mathcal{E}$ if there exists a subset of entities $\mathcal{E}' \subseteq \mathcal{E}$ with the properties that $\epsilon \in \mathcal{E}$, that $|\mathcal{E}'| \geq k$ and that $\mathcal{E}'$ is d-convergent.*

An important feature of this anonymity definition is that it provides anonymity guarantees that can be derived from a subset of all available data, but continue to hold once we consider a larger part of the dataset.

Corollary 3. *If an entity is (k, d)-anonymous in a collection of entities $\mathcal{E}$, then it is also (k, d)-anonymous in the collection of entities $\mathcal{E}' \supset \mathcal{E}$.*

Intuitively, (k, d)-anonymity is a generalization of the classical notions of k-anonymity to open settings without pre-defined quasi-identifiers. We schematically illustrate such anonymous subsets in Fig. 1.

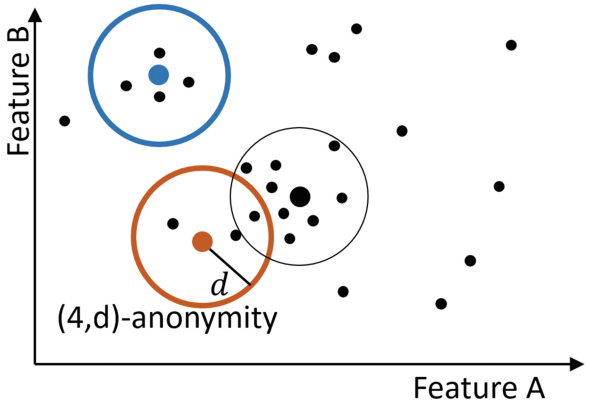


Fig. 1. Anonymity in crowdsourcing systems. (Color figure online)

5.3 Entity Matching

We define the notion of *matching* identities. As before, we use the distance measure dist to assess the similarity of two entities.

Definition 19. *An entity ϵ c-matches an entity ϵ' if* $\mathsf{dist}(\theta_\epsilon, \theta_{\epsilon'}) \leq c$.

Similarly, we can also define the notion of one entity matching a collection of entities.

Definition 20. *A collection of entities $\mathcal{E}$ c-*matches *an entity ϵ' if all entities $\epsilon \in \mathcal{E}$ c-match ϵ'.*

Assuming the adversary only has access to the similarity of entities, the best he can do is comparing the distance of all entities $\epsilon \in \mathcal{E}$ to ϵ' and make a probabilistic choice proportional to their relative distance values.

Now, if the matching identity ϵ^* is d-convergent in $\mathcal{E}$ the, all entities in $\mathcal{E}$ will have a comparatively similar distance to ϵ'.

Lemma 1. *Let $\mathcal{E}$ be d-convergent for ϵ^*. If ϵ^* c-matches ϵ', then $\mathcal{E}$ $(c+d)$-matches ϵ'.*

Proof. Since $\mathcal{E}$ is d-convergent for ϵ^*, $\forall \epsilon' \in \mathcal{E} : \mathsf{dist}(\epsilon^*, \epsilon') \leq d$. Using the triangle inequality, and the fact that ϵ^* c-matches the entity ϵ', we can bound the distance of all entities $\epsilon \in \mathcal{E}$ to ϵ' by $\forall \epsilon'' \in \mathcal{E} : \mathsf{dist}(\epsilon, \epsilon') \leq c + d$. Hence $\mathcal{E}$ $(c+d)$-matches the entity ϵ'. □

Hence, the matching entity ϵ^* does not c-match ϵ' for a small value of c, the adversary Adv he will have a number of possibly matching entities that are similarly likely to match ϵ'.

We get the same result if not the whole collection $\mathcal{E}$ is convergent, but if there exists a subset of convergent entities that allows the target to remain anonymous.

Corollary 4. *Let ϵ' be (k,d)-anonymous in $\mathcal{E}$. If ϵ' c-matches an entity ϵthen there is a subset $\mathcal{E}' \subseteq \mathcal{E}$ of size at least k which $(c+d)$-matches ϵ.*

5.4 Identity Disclosure

We assume that the adversary uses the similarity of the candidate entities to his target entity ϵ' to make his decision. The likelihood that the adversary chooses a specific entity ϵ^* then is the relative magnitude of $\mathsf{dist}(\epsilon^*, \epsilon)$, i.e.

$$Pr[\mathsf{Adv} \text{ chooses } \epsilon^*] = 1 - \frac{\mathsf{dist}(\epsilon^*, \epsilon')}{\sum_{\epsilon \in \mathcal{E}} \mathsf{dist}(\epsilon, \epsilon')}.$$

We can now bound the likelihood with which a specific entity ϵ^* would be chosen by the adversary if ϵ^* is (k,d)-anonymous.

Theorem 2. *Let the matching entity ϵ^* of the entity ϵ' in the collection $\mathcal{E} = \{\epsilon_1, \ldots, \epsilon_n\}$ be (k,d)-anonymous in $\mathcal{E}$. Furthermore let ϵ^* c-match ϵ'. Then an adversary $\mathsf{Adv} = (b, \emptyset)$ with uniform prior belief b and with empty world knowledge that only observes the similarity of entities links the entity ϵ^* to ϵ' with a likelihood of at most $t \leq 1 - \frac{c}{c+(k-1)(c+d)}$.*

Proof. Let $\mathcal{E}^*$ be the (k,d) anonymous subset of ϵ^* in $\mathcal{E}$. Let t^* be the likelihood of identifying ϵ^* from $\mathcal{E}^*$. Then clearly $t < t^*$ since we remove all possible, but wrong candidates in $\mathcal{E} \setminus \mathcal{E}^*$.

Since ϵ^* c-matches ϵ', by Lemma 1, we can upper bound the distance of each entity in $\mathcal{E}^*$ to ϵ', i.e.,

$$\forall \epsilon \in \mathcal{E}^* : \mathsf{dist}(\epsilon, \epsilon') \leq c + d$$

We can now bound t^* as follows:

$$\begin{aligned} t^* &= Pr[\mathsf{Adv} \text{ chooses } \epsilon] \\ &= 1 - \frac{c}{c + (k-1)(\sum\limits_{\epsilon \in \mathcal{E}^* \setminus \{\epsilon^*\}} \mathsf{dist}(\epsilon, \epsilon'))} \leq 1 - \frac{c}{c + (k-1)(c+d)} \end{aligned}$$

□

Theorem 2 shows that, as long as entities remain anonymous in a suitably large anonymous subset of a collection of entities, an adversary will have difficulty identifying them with high likelihood. Recalling our unlinkability definition from the beginning of the section, this result also implies that ϵ^* is σ-unlinkable for $\sigma = t$.

Corollary 5. *Let the matching entity ϵ^* of the entity ϵ' in the collection $\mathcal{E} = \{\epsilon_1, \ldots, \epsilon_n\}$ be (k,d)-anonymous in $\mathcal{E}$. Then ϵ^* and ϵ' are σ-unlinkable for $\sigma = 1 - \frac{c}{c+(k-1)(c+d)}$ against an adversary $\mathsf{Adv} = (b, \emptyset)$ with uniform prior belief and empty world knowledge that only observes entity similarity.*

In Sect. 6.5 we present experiments that evaluate the anonymity and linkability of individuals in the Online Social Network Reddit, and measure how well they can be identified from among their peers.

5.5 Limitations

The quality of the assessment provided by the d-convergence model largely depends on adversarial prior belief we assume: in our results above, we assume an adversary without any prior knowledge. In practice, however, the adversary might have access to prior beliefs that can help him in his decision making. Therefore, turning such assessments into meaningful estimates in practice requires a careful estimation of prior knowledge by, e.g., producing a more accurate profile model: the problem of comprehensive profile building for entities in an open setting is an open question that has been examined somewhat in the literature [8,13,17,19,53], but on the whole still leaves a lot of space for future work.

This concludes the formal definitions of our d-convergence model. In the next sections, we instantiate it for identity disclosure risk analyses based on user-generated text-content and apply this instantiation to the OSN Reddit.

6 Linkability Evaluation on Reddit

While the main focus of this paper is to present the actual privacy model as such, the following experiments are meant to provide first insights into the application of our framework, without taking overly complex adversarial capabilities into account. The evaluation can easily be extended to a more refined model of an adversary without conceptual difficulties.

We first articulate the goals of this evaluation, and then, secondly, describe the data collection process, followed by defining the instantiation of the general framework we use for our evaluation in the third step. Fourth, we introduce the necessary processing steps on our dataset, before we finally discuss the results of our evaluation.

6.1 Goals

In our evaluation, we aim at validating our model by conducting two basic experiments. First, we want to empirically show that, our model instantiation yields a suitable abstraction of real users for reasoning about their privacy. To this end, profiles of the same user should be more similar to each other (less distant) than profiles from different users.

Second, we want to empirically show that a larger anonymous subset makes it more difficult for an adversary to correctly link the profile. Thereby, we inspect whether anonymous subsets provide a practical estimate of a profile's anonymity.

Given profiles with anonymous subsets of similar size, we determine the percentage of profiles which the adversary can match within the top k results, i.e., given a source profile, the adversary computes the top k most similar (less distant) profiles in the other subreddit. We denote this percentage by *precision@k* and correlate it to the size of the anonymous subsets.

We fix the convergence of the anonymous subsets to be equal to the matching distance between two corresponding profiles. Our intuition is that, this way, the anonymous subset captures most of the profiles an adversary could potentially consider matching.

6.2 Data-Collection

For the empirical evaluation of our privacy model, we use the online social network Reddit [1] that was founded in 2005 and constitutes one of the largest discussion and information sharing platforms in use today. On Reddit, users share and discuss topics in a vast array of topical subreddits that collect all topics belonging to one general area; e.g. there are subreddits for world news, tv series, sports, food, gaming and many others. Each subreddit contains so-called submissions, i.e., user-generated content that can be commented on by other users.

To have a ground truth for our evaluation, we require profiles of the same user same user across different OSNs to be linked. Fortunately, Reddit's structure provides an inherent mechanism to deal with this requirement. Instead of considering Reddit as a single OSN, we treat each subreddit as its own OSN. Since users are identified through the same pseudonym in all of those subreddits, they remain linkable across the subreddits' boundaries. Hence our analysis has the required ground truth. The adversary we simulate, however, is only provided with the information available in the context of each subreddit and thus can only try to match profiles across subreddits. Ground truth in the end allows us to verify the correctness of his match.

To build up our dataset, we built a crawler using Reddit's API to collect comments. Recall that subreddits contain submissions that, in turn, are commented by the users. For our crawler, we focused on the large amount of comments because they contain a lot of text and thus are best suitable for computing the unigram models.

Our crawler operates in two steps that are repeatedly executed over time. During the whole crawling process, it maintains a list of already processed users. In the first step, our crawler collects a list of the overall newest comments on Reddit from Reddit's API and inserts these comments into our dataset. In the second step, for each author of these comments who has not been processed yet, the crawler also collects and inserts her latest $1,000$ comments into our dataset. Then, it updates the list of processed users. The number of $1,000$ comments per user, is a restriction of Reddit's API.

In total, during the whole September 2014, we collected more than 40 million comments from over $44,000$ subreddits. The comments were written by about $81,000$ different users which results in more than 2.75 million different profiles.

The whole dataset is stored in an anonymized form in a MySQL database and is available upon request.

6.3 Model Instantiation

On Reddit, users only interact with each other by by posting comments to text of link submissions. Reddit therefore does not allow us to exploit features found in other social networks, such as friend links or other static data about each user. On the other hand, this provides us with the opportunity to evaluate the

linkability model introduced in Sect. 5 based dynamic, user-generated content, in this case user-generated text content.

Since we only consider text content, we instantiate the general model from the previous sections with an unigram model, where each attribute is a word unigram, an its value is the frequency with which the unigram appears in the profiles comments. Such unigram models have succesfully been used in the past to characterize the information within text content and to correlate users across different online platforms [28,45].

Definition 21 (Unigram Model). *Let $\mathcal{V}$ be a finite vocabulary. The* unigram model $\theta_P = p_i$ *of a profile is a set of* frequencies $p_i \in [0, \ldots, 1]$ *with which each unigram $w_i \in \mathcal{V}$ appears in the profile P. Each frequency p_i is determined by*

$$p_i = \frac{\mathsf{count}(w_i, P)}{\sum_{w \in \mathcal{V}} \mathsf{count}(w, P)}$$

Since the unigram model essentially constitutes a probability distribution, we instantiate our distance metric dist with the Jensen-Shannon divergence [25]. The Jensen-Shannon divergence is a symmetric extension of the Kullback-Leiber divergence has been shown to be successful in many related information retrieval scenarios.

Definition 22. *Let P and Q be two statistical models over a discrete space Ω. The Jensen-Shannon divergence is defined by*

$$D_{\mathsf{JS}} = \frac{1}{2} D_{\mathsf{KL}}(P||M) + \frac{1}{2} D_{\mathsf{KL}}(Q||M)$$

where D_{KL} is the Kullback-Leibler divergence

$$D_{\mathsf{KL}}(P||Q) = \sum_{\omega \in \Omega} log\left(\frac{P(\omega)}{Q(\omega)}\right) P(\omega)$$

and M is the averaged distribution $M = \frac{1}{2}(P + Q)$.

In the following, we will use the square-root of the Jensen-Shannon divergence, constituting a metric, as our distance measure, i.e., $\mathsf{dist} = \sqrt{D_{\mathsf{JS}}}$.

6.4 Data-Processing

The evaluation on our dataset is divided into sequentially performed computation steps, which include the normalization of all comments, the computation of unigram models for each profile, a filtering of our dataset to keep the evaluation tractable, the computation of profile distances and the computation of (k, d)-anonymous subsets.

Normalizing Comments. Unstructured, heterogeneous data, as in our case, may contain a variety of valuable information about a user's behavior, e.g., including formatting and punctuation. Although we could transform these into attributes, we do not consider them here for the sake of simplicity.

In order to get a clean representation to apply the unigram model on, we apply various normalization steps, including transformation to lower case, the removal of Reddit formatting and punctuation except for smilies. Moreover, we apply a encoding specific normalization, replace URLs by their hostnames and shorten repeated characters in words like `cooool` to a maximum of three. Finally, we also filter out a list of 597 stopwords from the comments. Therefore, we perform six different preprocessing steps on the data, which we describe in more detail in the following.

1. **Convert to lower case letters:** In our statistical language models, we do not want to differentiate between capitalized and lowercased occurrences of words. Therefore, we convert the whole comment into lower case.
2. **Remove Reddit formatting:** Reddit allows users to use a wide range of formatting modifiers that we divide into two basic categories: formatting modifiers that influence the typography and the layout of the comment, and formatting modifiers that include external resources into a comment. The first kind of modifier, named layout modifiers, is stripped off the comment, while leaving the plain text. The second kind of modifier, called embedding modifiers, is removed from the comment completely.
   One example for a layout modifier is the asterisk: When placing an asterisk both in front and behind some text, e.g., `*text*`, this text will be displayed in italics, e.g., *text*. Our implementation removes these enclosing asterisks, because they are not valuable for computing statistical language models for n-grams and only affect the layout. Similarly, we also remove other layout modifiers such as table layouts, list layouts and URL formatting in a way that only the important information remains.
   A simple example for embedding modifiers are inline code blocks: Users can embed arbitrary code snippets into their comments using the ‘ modifier. Since these code blocks do not belong to the natural language part of the comment and only embed a kind of external resource, we remove them completely. In addition to code blocks, the category of embedding modifiers also includes quotes of other comments.
3. **Remove stacked diacritics:** In our dataset, we have seen that diacritics are often misused. Since Reddit uses Unicode as its character encoding, users can create their own characters by arbitrarily stacking diacritics on top of them. To avoid this kind of unwanted characters, we first normalize the comment by utilizing the unicode character composition, which tries to combine each letter and its diacritics into a single precombined character. Secondly, we remove all remaining diacritic symbols from the comment. While this process preserves most of the normal use of diacritics, it is able to remove all unwanted diacritics.

4. **Replace URLs by their hostname:** Generally, a URL is very specific and a user often does not include the exact same URL in different comments. However, it is much more common that a user includes different URLs that all belong to the same hostname, e.g., www.mypage.com. Since our statistical language models should represent the expected behavior of a user in terms of used words (including URLs), we restrict all URLs to their hostnames.
5. **Remove punctuation:** Most of the punctuation belongs to the sentence structure and, thus, should not a part of our statistical language models. Therefore, we remove all punctuation except for the punctuation inside URLs and smilies. We do not remove the smilies, because people are using them in a similar role as words to enrich their sentences: Every person has her own subset of smilies that she typically uses. To keep the smilies in the comment, we maintain a list of 153 different smilies that will not be removed from the comment.
6. **Remove duplicated characters:** In the internet, people often duplicate characters in a word to add emotional nuances to their writing, e.g., `cooooo-oooool`. But sometimes the number of reduplicated characters varies, even if the same emotion should be expressed. Thus, we reduce the number of duplicated characters to a maximum of 3, e.g., `coool`. In practice, this truncation allows us to differentiate between the standard use of a word and the emotional variation of it, while it does not depend on the actual number of duplicated characters.

Computing Unigram Models. From the normalized data, we compute the unigram frequencies for each comment. Recall that our dataset consists of many subreddits that each form their own OSN. Thus, we aggregate the corresponding unigram frequencies per profile, per subreddit, and for Reddit as a whole. Using this data, we compute the word unigram frequencies for each comment as described in Sect. 6.3.

Since a subreddit collects submissions and comments to a single topic, we expect the unigrams to reflect its topic specific language. Indeed, the 20 most frequently used unigrams of a subreddit demonstrate that the language adapts to the topic. As an example, we show the top 20 unigrams (excluding stopwords) of Reddit and two sample subreddits *Lost* and *TipOfMyTongue* in Table 1. As expected, there are subreddit specific unigrams that occur more often in the context of one subreddit than in the context of any other subreddit. For example, the subreddit *Lost* deals with a TV series that is about the survivors of a plane crash and its aftermath on an island. Unsurprisingly, the word *island* is the top unigram in this subreddit. In contrast, the subreddit *TipOfMyTongue* deals with the failure to remember a word from memory and, thus, has the word *remember* in the list of its top three unigrams.

Filtering the Dataset. To reduce the required amount of computations we restrict ourselves to *interesting profiles*. We define an interesting profile as one that contains at least 100 comments and that belongs to a subreddit with at

Table 1. Top 20 unigrams of Reddit and two sample subreddits Lost and TipOfMyTongue.

Top	Reddit		Subreddit: Lost		Subreddit: TipOfMyTongue	
	Unigram	Frequency	Unigram	Frequency	Unigram	Frequency
1	people	4,127,820	island	832	www.youtube.com	3663
2	time	2,814,841	show	750	song	1,542
3	good	2,710,665	lost	653	remember	1,261
4	gt	2,444,240	time	580	en.wikipedia.org	1,100
5	game	1,958,850	people	527	sounds	1,007
6	pretty	1,422,640	locke	494	solved	924
7	2	1,413,118	season	431	movie	918
8	lot	1,385,167	jacob	429	find	829
9	work	1,352,292	mib	372	:)	786
10	1	1,184,029	jack	310	game	725
11	3	1,124,503	episode	280	time	678
12	great	1,070,299	ben	255	thinking	633
13	point	1,063,239	good	250	good	633
14	play	1,060,985	monster	237	www.imdb.com	584
15	years	1,032,270	lot	220	video	583
16	bad	1,008,607	gt	182	pretty	570
17	day	989,180	character	165	youtu.be	569
18	love	988,567	walt	163	mark	548
19	find	987,171	man	162	edit	540
20	shit	976,928	dharma	162	post	519

least 100 profiles. Additionally, we dropped the three largest subreddits from our dataset to speed up the computation.

In conclusion, this filtering results in 58,091 different profiles that belong to 37,935 different users in 1,930 different subreddits.

Distances Within and Across Subreddits. Next, we compute the pairwise distance within and across subreddits using our model instantiation. Excluding the distance of profiles to themselves, the minimal, maximal and average distance of two profiles within subreddits in our dataset are approximately 0.12, 1 and 0.79 respectively. Across subreddits, the minimal, maximal and average distance of two profiles are approximately 0.1, 1 and 0.85 respectively.

Anonymous Subsets. Utilizing the distances within subreddits, we can determine the anonymous subsets for each profile in a subreddit. More precisely, we compute the anonymous subset for each pair of profiles from the same user.

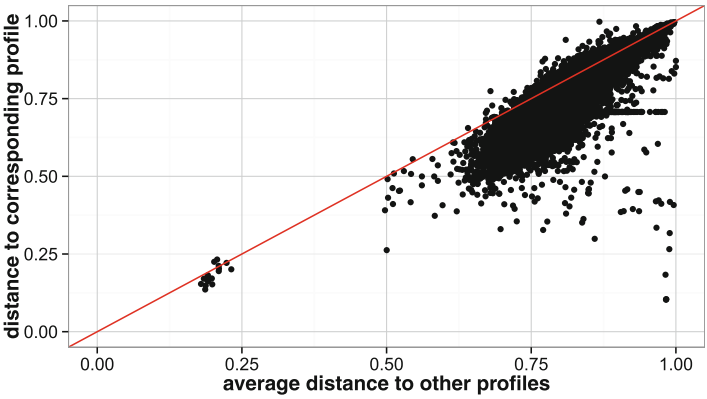


Fig. 2. The average distance between a profile in subreddit s and all profiles in s' versus the matching distance between the profile and its correspondence in s'. (Color figure online)

We set the convergence d to the matching distance between both profiles and determine the size of the resulting anonymous subset.

6.5 Evaluation and Discussion

In this subsection, we inspect and interpret the results of our experiments with regard to our aforementioned goals. Therefore, we first start by giving evidence that our approach indeed provides a suitable abstraction of real users for reasoning about their privacy.

To this end, we compare the distance of matching profiles to the average distance of non-matching profiles. In particular, for each pair of profiles from the same user in subreddits s and s', we plot the average distance from the profile in s to the non-matching profiles in s' in relation to the distance to the matching profile in s' in Fig. 2. The red line denotes the function $y = x$ and divides the figure into two parts: if a point lies below the line through the origin, the corresponding profiles match better than the average of the remaining profiles. Since the vast majority of datapoints is located below the line, we can conclude that profiles of the same user match better than profiles of different users.

Our second goal aimed at showing that anonymous subsets indeed can be used to reason about the users' privacy. Therefore, we investigate the chances of an adversary to find a profile of the same user within the top k matches and relate its chance to the size of the profile's anonymous subset. More precisely, given multiple target profiles with similar anonymous subset sizes, we determine the, so called, precision@k, i.e., the ratio of target profiles that occur in the top k ranked matches (by ascending distance from the source profiles). We relate this precision@k to the anonymous subset sizes with a convergence d set to the

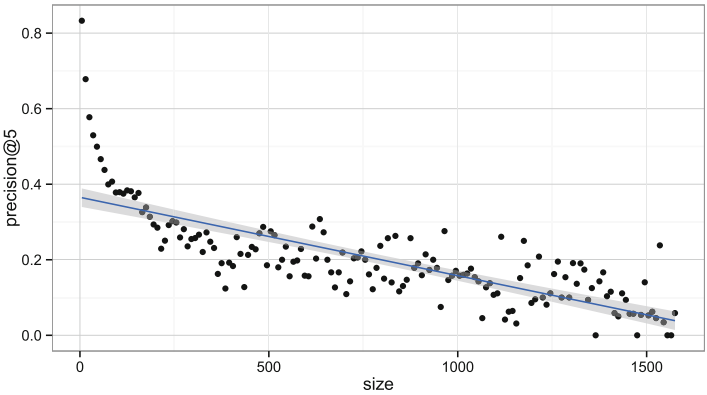


Fig. 3. The anonymous subset size correlated to the precision an adversary has if considering the top 5 profiles as matching. (Color figure online)

distance between the source and target profiles, and we group the anonymous subset sizes in bins of size 10.

In our evaluation, we considered $k \in \{1, 5, 10, 20\}$, which all yield very similar results. Exemplarily, we correlate the aforementioned measures for $k = 5$ in Fig. 3, clearly showing that an increasing anonymous subset size correlates with an increasing uncertainty – i.e., decreasing precision – for the adversary.

7 Assessing the Effectiveness of Countermeasures Against Authorship Recognition

In this section, we explore another application of the linkability model we introduced in Sect. 5: we develop a method to assess the effectiveness of various countermeasures against authorship recognition, i.e. the process of linking text content that were authored by the same user based on stylometric features exhibited by the content.

7.1 Theoretical Foundation

We first develop the formal foundation for our evaluation of authorship recognition countermeasures. We derive our definitions from those in the previous sections and adapt them to capture information about writing style.

Threat Model. In our threat model, we assume multiple collections of entities $\mathcal{E}_i$, also called *communities*. An entity $\epsilon \in \mathcal{E}_i$ is characterized by its writing style and corresponds to a pseudonymous author of a collection of texts. Two entities ϵ_1 and ϵ_2 are called matching, if both belong to the same author.

The adversary's goal is now to identify matching entities across several communities by analyzing their writing style. Figure 4 (see Sect. 7.1) shows two exemplary communities including the links between matching entities.

For the application of countermeasures, we assume that a community $\mathcal{E}$ already exists, together with all text passages published by the entities in $\mathcal{E}$, and a test author applies a countermeasure on his text passage before it is published into $\mathcal{E}$. We simulate this by simply choosing a subset of test authors from $\mathcal{E}$ for which we evaluate the countermeasures.

Statistical Models of Writing-Style. For authorship attribution, we extend the definition of entity models from Definition 1 in Sect. 4 to include different types of attributes that each will represent one feature type of the writing style feature set (e.g., as presented in [3]).

Definition 23 (Attribute Types). *An* attribute type class $\mathcal{T}$ *is a collection of* attribute types $\tau \in \mathcal{T}$. *We denote with* $\mathcal{A}_\tau$ *the set of all attributes* $\alpha \in \mathcal{A}_\tau$ *that realize the attribute type* τ.

Intuitively, an attribute type corresponds to a feature or class of features, e.g., the sentence length or word unigrams. A possible realization for the sentence length would be 5, whereas *house* is a possible realization for a word unigram. Statistical models now associate with each attribute and attribute type a probability, or relevance estimation, of this attribute for the specific entity.

Definition 24 (Extended Statistical Model). *The* entity model $\theta_\epsilon = (\theta_\epsilon^{\tau_1}, \ldots, \theta_\epsilon^{\tau_n})$ *of an entity* ϵ *consists of the statistical models of its attribute types* $\tau_i \in \mathcal{T}, 1 \leq i \leq n$.

Each statistical model $\theta_\epsilon^{\tau_i}$ *determines the probability* $Pr[\alpha \mid \theta_\epsilon^{\tau_i}]$ *that the entity* ϵ *exhibits the attribute* $\alpha \in \mathcal{A}_{\tau_i}$.

In the easiest case, the probability of exhibiting a specific attribute (i.e., a specific feature realization) will be proportional to its frequency in a user's text. Additionally, in our experimental evaluation, we also explore the possibility to set the probability proportional to the popular *term frequency inverse document frequency* to better capture the relevance of an attribute in a user's text given the context in which it is published.

Entity Similarity. In the following, we will use these entity models together with a similarity measure on these models to evaluate the similarity of entities with regard to their writing style. We follow the intuition that a higher similarity between two entities in different communities implies a higher likelihood that they both correspond to the same author.

Since we characterize each author in terms of statistical models, determining their similarity boils down to measuring the similarity of probability distributions. As proposed in Sect. 6.3, we utilize the Jensen-Shannon divergence [25] to

determine the similarity of our statistical models. The Jensen-Shannon divergence is a symmetric variant of the popular Kullback-Leibler divergence, and fulfills all properties of a metric distance measure.

We extend this similarity measure to fit our notion of statistical models with attribute types, resulting in a linear combination of the similarities of each attribute type.

Definition 25 (Similarity of Entities). *Given two entities $\epsilon_1, \epsilon_2 \in \mathcal{E}$, the similarity of ϵ_1 and ϵ_2 is the linear combination of the similarities of their statistical models. Let $\mathbf{sim}(\theta_{\epsilon_1}, \theta_{\epsilon_2}) = (\mathsf{sim}(\theta^{\tau_1}_{\epsilon_1}, \theta^{\tau_1}_{\epsilon_2}), \ldots, \mathsf{sim}(\theta^{\tau_n}_{\epsilon_1}, \theta^{\tau_n}_{\epsilon_2}))$ and $\boldsymbol{\lambda} = (\lambda_{\tau_1}, \ldots, \lambda_{\tau_n})$. Then,*

$$\mathsf{sim}(\theta_{\epsilon_1}, \theta_{\epsilon_2}) = \boldsymbol{\lambda} \cdot \mathbf{sim}(\theta_{\epsilon_1}, \theta_{\epsilon_2}) + \rho,$$

where ρ denotes an optional constant.

When applying this theory to an actual dataset, $\boldsymbol{\lambda}$ and ρ can be learned using established regression and classification techniques.

Average Entity in a Collection. Next, we further extend statistical models to a collection of entities $\mathcal{E}$, which gives us the probability that a randomly chosen entity from the collection exhibits an attribute. While the former part of this section introduced the formal ground for attributing authorship by similarities, the definitions in this and the next paragraph will be used for powering some of our countermeasures.

Definition 26 (Stat. Models for Collections). *Given a set of attribute types $\mathcal{T}$, the statistical model $\theta_{\mathcal{E}}$ of a collection of entities $\mathcal{E}$ is defined as $(\theta^{\tau_1}_{\mathcal{E}}, \ldots, \theta^{\tau_n}_{\mathcal{E}})$, where each $\theta^{\tau_i}_{\mathcal{E}}$ determines the probability $Pr[\alpha \mid \theta^{\tau_i}_{\mathcal{E}}]$ that an entity $\epsilon \in \mathcal{E}$, chosen uniformly at random, exhibits an attribute $\alpha \in \mathcal{A}_{\tau_i}$.*

We can compute each statistical model $\theta^{\tau_i}_{\mathcal{E}}$ of a collection $\mathcal{E}$ by

$$Pr[\alpha \mid \theta^{\tau_i}_{\mathcal{E}}] = \frac{\sum_{\epsilon \in \mathcal{E}} Pr[\alpha \mid \theta^{\tau_i}_{\epsilon}]}{|\mathcal{E}|}$$

for each attribute $\alpha \in \mathcal{A}_{\tau_i}$.

The statistical model for a collection corresponds to the average entity in that collection.

(k, d)-anonymity. As described in Sect. 5.2 (cf. Definition 18), we assess anonymity of an entity by identifying *anonymous* subsets within a community that allow an entity to hide amongst her peers: The (k, d)-anonymous subset of an entity $\epsilon \in \mathcal{E}$ is a subset of entities $\mathcal{A} \subseteq \mathcal{E}$ of size k, each of which are at least d-similar to ϵ. For a fixed value of k, the anonymous subset's convergence is a good indicator for *how* close the nearest k entities are. We will utilize these anonymous subsets to improve the automatic countermeasures we propose in Sect. 7.2 by not changing the text towards the average author from the whole community, but rather an existing author within an anonymous subset of the community.

Countermeasure Formalization. Finally, we formally define countermeasures in the context of statistical models and then define our novel notion of gain provided by a countermeasure.

Definition 27 (Countermeasure). *A* countermeasure $\mathcal{C}$ *is a function that changes the statistical model* θ_ϵ *to* $\mathcal{C}(\theta_\epsilon)$.

The optimal weights $\boldsymbol{\lambda} = \lambda_{\tau_1}, \ldots, \lambda_{\tau_n}$ obtained from the regression or trained classifier can be used to determine the importance of each attribute type for the stylometric similarity. Since their values might also be negative, the actual importance is defined as $(\lambda_{\tau_i})^2$, similar to an approach by Guyon *et al.* [29].

Definition 28 (Feature Importance). *Given* $\boldsymbol{\lambda}$ *and an attribute type* τ, τ*'s importance is defined as* $(\lambda_\tau)^2$. *The vector* $\boldsymbol{\mathcal{I}}$ *is defined as the element-wise multiplication* $\boldsymbol{\lambda} \odot \boldsymbol{\lambda}$ *and contains each attribute type's importance.*

In an ideal, private world no attribute type reliably contributes to the matching of corresponding authors, and hence no attribute type is particularly important. We capture this ideal scenario through ideal importances $\hat{\boldsymbol{\mathcal{I}}}$ that we aim to achieve through the application of countermeasures. In our case, if no attribute type is particularly important, we set $\hat{\boldsymbol{\mathcal{I}}} = \mathbf{0}$. Motivated by this intuition, we define a countermeasure's gain as the improvement towards the ideal scenario.

Definition 29 (Gain). *Let* $\boldsymbol{\mathcal{I}}$ *be attribute type importances before the application of a countermeasure and* $\boldsymbol{\mathcal{I}}'$ *be attribute type importances after the application. Then the* improvement potential *towards the ideal scenario* $\hat{\boldsymbol{\mathcal{I}}}$ *is defined as* $\boldsymbol{\mathcal{I}} - \hat{\boldsymbol{\mathcal{I}}}$.

A countermeasure $\mathcal{C}$*'s* gain $\mathsf{gain}_{\mathcal{C}}^{\tau}$ *with respect to a specific attribute type* τ *is the actual improvement towards the ideal scenario, while the countermeasure's overall gain* $\mathit{gain}_{\mathcal{C}}$ *is defined as the sum over the gains for all attribute types.*

$$\mathsf{gain}_{\mathcal{C}}^{\tau} = |\boldsymbol{\mathcal{I}}_\tau - \hat{\boldsymbol{\mathcal{I}}}_\tau| - |\boldsymbol{\mathcal{I}}'_\tau - \hat{\boldsymbol{\mathcal{I}}}_\tau|$$

$$\mathsf{gain}_{\mathcal{C}} = \sum_{\tau \in \mathcal{T}} \mathsf{gain}_{\mathcal{C}}^{\tau}$$

In the case that the ideal importances are 0*, this simplifies to* $\mathsf{gain}_{\mathcal{C}}^{\tau} = \boldsymbol{\mathcal{I}}_\tau - \boldsymbol{\mathcal{I}}'_\tau$.

Comparison to Other Measures. We compare our approach of computing the gain of countermeasures to other approaches that can be used to capture the effectiveness of countermeasures. Namely, we consider both (1) classifier-dependent measures such as precision, recall, accuracy, and (2) the classifier-independent measure of information gain.

By comparing such measures before and after the countermeasure's application, a similar measure to our gain is achieved. In contrast to our approach, however, both of the above approaches lead to drawbacks we will elaborate in the following.

Classifier-Dependent Measures. In general, a comparison of precision, recall and accuracy before and after a countermeasure's application only gives a global view on the effectiveness of a countermeasure, i.e., the global loss in those measures after the countermeasure's application. Such an approach fails in giving precise results on a feature-class level, since the underlying measures describe the total outcome of the classification.

Classifier-Independent Measures. While information gain is capable of both, providing a feature-level assessment of importance as well as being classifier-independent, it still fails to match our needs: Intuitively, a feature's information gain is higher if it is more discriminating. However, in its computation, information gain does not take into account which authors are actually matching.

Narayanan *et al.* [46] define information gain as $IG(F_i) = H(B) - H(B \mid F_i) = H(B)+H(F_i)-H(B, F_i)$ where H is the Shannon entropy, B is the random variable corresponding to a set identifier (in their case, the blog number), and F_i is the random variable corresponding to feature i. Adopting this definition let us define the notions of a feature's information gain for authors $IG_A(F_i)$ and for entities $IG_\epsilon(F_i)$.

Unfortunately, knowing which features distinguish authors is not necessarily the same as knowing which features help matching authors. For example, consider the two communities in Fig. 4, where matching entities are highlighted in the same color and are connected by a line. In this scenario, $IG_A(F_i)$ tells us to which extent F_i helps distinguishing the authors in general. However, without considering the boundaries between both communities, it is possible that F_i is only well discriminating in C_1, but not in C_2, and is in particular not very helpful in matching from C_1 to C_2.

Using $IG_\epsilon(F_i)$ instead respects the boundaries of the communities, but in fact does not help us in matching the entities across both communities, since $IG_\epsilon(F_i)$ only tells us which features can be used to distinguish between all entities within one community. The same feature might very well be completely useless in discriminating entities in the other community.

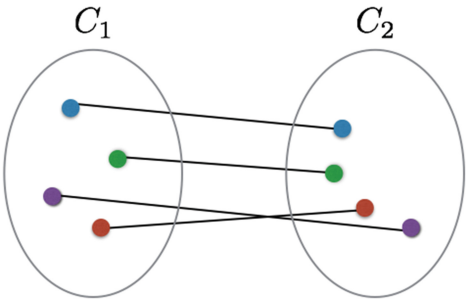


Fig. 4. Two different communities with matching authors. (Color figure online)

Gain. In contrast to the previous methods, gain is directly defined in terms of the optimal matching and values each feature class in their importance for achieving this matching. In the rest of the paper, we will not only show the validity of our approach by correlating the countermeasures' gain to classifier-dependent performance measures, but we will also demonstrate the usefulness of our methodology in a detailed analysis of several countermeasures.

7.2 Experimental SetUp

This section provides an overview over our experimental setup. In particular, we provide a detailed explanation of our dataset, the stylometric features we consider and the countermeasures we evaluate, including a description of each countermeasures' implementation.

Dataset. For our experimental evaluation, we leverage the Extended-Brennan-Greenstadt corpus [10,11], which provides a decent collection of writing samples from 45 different authors. The corpus contains writing samples of at least 6500 words for each author, which are split into approximately 500-word passages. Each writing sample is from a formal source, e.g., school essays, reports or other types of professional or academic correspondence, which was manually validated by the creators of the corpus.

We do not use the (1) *obfuscation* and (2) *imitiation* part of the corpus, in which the authors were requested to write passages on a specific topic while (1) trying to hide their writing style and (2) trying to imitate the writing style of another author, namely Cormac McCarthy. The methodology we develop is intended for evaluating the effect of countermeasures changing a given text, while both the obfuscation and imitation part of the corpus are already obfuscated texts that do not correspond to the original writing samples.

In order to evaluate authorship attribution between different communities, we artifically distribute all text passages of an author into three distinct communities. This way, our dataset consists of 3 communities, each containing 233 text passages of 45 authors (between 4 to 8 text passages per author in one community).

Feature Set. For our evaluation, we take the Writeprints extended feature set [3] as a basis that we further extend with additional features. However, we remove *word trigrams* to make our evaluation computationally more tractable. In total, we include 33 different stylometric features into our model, some of which we adjust to fit the structure of our dataset.

In correspondence with the model presented in Sect. 7.1, a feature class, such as, e.g., letter bigrams, corresponds to an attribute type, whereas each instance of a feature, e.g., the actual letter bigram "aa", corresponds to an attribute within this attribute type. During the evaluation, we store the frequency of each attribute and construct the statistical model from this observation. The quantity of a feature class describes the maximum number of features in that feature class, which we observed in our unmodified dataset. As mentioned in

Table 2. List of features.

Category	Identifier	Description	Quantity	λ	_spell	_syn	_mis	_spch
Character-Based	F1	count of letters (e.g., a, b, c)	44	−0.00936	✓	✓	✓	✓
	F2	letter bigrams (e.g., aa, ab, ac)	634	0.00133	✓	✓	✓	✓
	F3	letter trigrams (e.g., aaa, aab, aac)	5048	−0.00304	✓	✓	✓	✓
	F4	digits (e.g., 1, 2, 3)	10	0.00161				
	F5	digit bigrams (e.g., 01, 11, 12)	100	0.00379				
	F6	digit trigrams (e.g., 011, 111, 112)	1000	0.00138				
	F7	frequency of punctuation determined by unicode punctuation categories	27	0.01102	✓		✓	
	F8	frequency of punctuation as in Writeprints (?!,.'\";:)	8	0.00688	✓		✓	
	F9	frequency of special characters determined unicode symbol character categories (except for modifier symbols)	6	−0.01028	✓			✓
	F10	frequency of special characters as in Writeprints (~@#$%^&*-_=+><[]{}/\\|)	19	−0.00906	✓			✓
	F11	number of characters per text passage	432	−0.00114	✓	✓	✓	✓
POS-Tagger-Based	F12	gunning fog index (rounded to nearest whole number)	23	0.00421	✓	✓	✓	✓
	F13	flesch reading ease (rounded to nearest tenth)	10	0.00042	✓	✓	✓	✓
	F14	flesch kincaid grade level (rounded to nearest whole number)	48	−0.00108	✓	✓	✓	✓
	F15	frequency of POS tags (e.g., NN, VB)	44	0.009	✓	✓	✓	✓
	F16	frequency of POS tag bigrams (e.g., NN VB)	1171	−0.03563	✓	✓	✓	✓
	F17	frequency of POS tag trigrams (e.g., NN VB NN)	12,507	0.00387	✓	✓	✓	✓
	F18	number of characters per sentence	388	0.00471	✓	✓	✓	✓
	F19	number of words per sentence	110	−0.00939	✓	✓	✓	✓
	F20	ratio of adjectives and adverbs compared to the total number of words	13	0.00149	✓	✓	✓	✓
	F21	ratio of comparatives and superlatives compared to all adjectives	16	0.00214	✓	✓	✓	✓
	F22	ratio of nouns to all words	22	0.00311	✓	✓	✓	✓
	F23	ratio of verbs to all words	14	0.00145	✓	✓	✓	✓
	F24	ratio of verbs in past tense to all verbs	69	0.00448	✓	✓	✓	✓
	F25	ratio of verbs in the third person present to all verbs	49	−0.00101	✓	✓	✓	✓
	F26	ratio of verbs in the first or second person to all verbs	76	0.00108	✓	✓	✓	✓
Word-Based	F27	hapax legomena	21,126	−0.04912	✓	✓	✓	✓
	F28	misspelled words (e.g., abandonned, abudance)	37	−0.00042	✓		✓	
	F29	frequency of function words (e.g., I, for, of)	451	0.00502	✓	✓	✓	
	F30	word unigrams (e.g., lemon, tree)	21,723	0.12246	✓	✓	✓	✓
	F31	word bigrams (e.g., lemon tree)	172,764	0.04693	✓	✓	✓	✓
	F32	number of words per text passage	64	0.00153	✓			✓
	F33	number of characters per word	30	−0.0058	✓	✓	✓	✓

Sect. 7.1, we also evaluate the use of *term frequency inverse document frequency* instead of frequency to instantiate our statistical models in Sect. 8.

A full list of the feature classes and their quantities can be found in Table 2. We will use the identifiers F1 to F33 for each feature class introduced in this table throughout the rest of the paper. The table's last five columns will be formally introduced throughout the next sections.

In general, we group our features into three different categories depending on the actual implementation:

1. *Character-based* feature classes, for which we represent an author's text-passage as a list of characters and compute the corresponding attribute frequencies on that list.
2. *POS-tagger-based* feature classes rely on the output of the Stanford POS tagger [55,56] used with a twitter model [20] in order to use enhanced information about the current sentence and word in a text passage (e.g., a word's POS tag).
3. *Word-based* feature classes leverage a Java break iterator to efficiently iterate over the words of an authors text passages.

Additional resources from which we construct feature classes include a syllable counter that first tries to determine a word's syllable count from the dictionary CMUDict [18] by counting the number of vowels in the pronunciation. In case of failure, it determines an approximate syllable count based on an algorithm written by Greg Fast [26], counting the number of vowel groups in the word and adjusting the number for certain special cases. Moreover, we use a list of 512 function words as well as a list of common misspellings taken from Wikipedia [58] and the Anonymouth framework [42] to construct feature classes (e.g., F28 and F29).

Countermeasures. In this section, we discuss the countermeasures whose impact we aim to evaluate. We detail their implementation and also argue why these countermeasures preserve the semantics of the text. Finally, we present a list of features affected by each countermeasure.

Generally, we distinguish between two types of countermeasures: *simple countermeasures* and *optimizing countermeasures*. Simple countermeasures apply the first possible action to a given text, independent of its context, whereas optimizing countermeasures rank each available action and apply the most promising one. We first introduce all countermeasures in general before we discuss their optimizing variants in Sect. 7.2.

In total, our experiments incorporate four different countermeasures, which we will not only apply individually, but also in meaningful combinations. For referencing purposes, we name our countermeasures and present the affected features per countermeasure in Table 2 (some of which can be affected indirectly by, e.g., causing the POS tagger to fail).

Spell Checking (`_spell`). Since we are interested in assessing the impact of a standard text rewriting tool on the anonymity of text authors, we start with

the arguably most common such tool: a spell checker. Spell checkers constitute a simple, but widely used example for tools that modify a text.

Our implementation of this countermeasure employs the open source Java spell checker `LanguageTool` [32] that is even able to detect grammar problems. Each text-passage that gets fed into this countermeasure will be corrected by the spell checker, always choosing the first suggestion. Due to the usual field of application of spell checkers, we consider this countermeasure to be mostly semantics-retaining.

Synonym Substitution (`_syn`). Our technically most sophisticated countermeasure replaces words by synonyms. Considering the highly flexible and changing nature of language, this task introduces several challenges:

1. Most often in natural language, words do not occur in its root form but rather in an inflected grammatical form. Thus, it is essential to get the canonical root form of a word.
2. Given such a root form, we have to maintain a dictionary of synonyms for each word. The dictionary should contain synonyms for at least nouns, verbs and adjectives – possibly also adverbs.
3. We need to be able to examine the inflection of the original word in order to replace it by a synonym in exactly the same grammatical form.
4. Finally, if we know the desired form, we have to inflect the synonym.

Fortunately, 1. and 2. can both be handled by leveraging WordNet [27,44], which is a large lexical database providing so called synsets for nouns, verbs, adjectives and adverbs. It can also be queried using non-root forms, which renders the first challenge irrelevant.

Since English as a language is only weakly inflected, a few hints for 3. suffice to generate the correct inflection of the synonym. More concretely, the output of the Stanford POS Tagger [55,56] used with a twitter model [20] is enough to infer the lexical category as well as attributes like tense, plural and person that allow us to determine the correct grammatical form.

Finally, we use simplenlg, a natural language generation API for Java, to realize the correct form of our synonym.

Repeating this for all potential synonyms, we select the optimal replacement according to our optimization strategy. Consequently, this countermeasure is optimizing. Also, substituting words by synonyms should preserve the initial semantics of a text.

Adding/Removing Misspellings (`_mis`). Another optimizing countermeasure that we implement makes use of misspellings, e.g., for cases where the misspelled word is more common than its correct form. Thus, this countermeasure first looks up the correct or misspelled variants of a word from a dictionary and then evaluates which form to use.

To accomplish this task, we adapt a list of common misspellings from Wikipedia [58] and generate a dictionary, providing a set of possibly misspelled and

corrected substitutions for every word in the list. Although a misspelled word can potentially create confusion, this kind of substitutions should not drastically change the semantics of a text, because the correct word is nearly always recognizable from the context.

Special Characters Modification (`_spch`). Our last countermeasure seeks for the replacement of potentially identifying special characters. Since special characters (excluding punctuation) generally occur only infrequently in natural language, their usage is more likely to be unique and, consequently, can lead to an easy author identification. In order to counter this problem, we created a list of the most common special characters in our dataset and mapped them to their textual meanings. One example for such a mapping is © ↔ `copyright` ↔ `(C)`, where each of the three alternatives could be substituted by another if the optimization yields a higher result.

Obviously, if the special characters are used with their usual meaning, this countermeasure preserves the semantics of a text.

Combinations. Instead of only evaluating the presented countermeasures individually, we also examine the impact of multiple countermeasures applied in a sequential manner. However, we exclude any combination that involves both `_spell` and `_mis` as the two countermeasures are very similar to each other and potentially could cancel each other out. To further narrow down the number of possible combinations, we also only apply countermeasures in a meaningful order. For example, it makes sense to apply spell checking first, because it is not optimizing and thus could influence the result of previous countermeasures in a negative way. Synonym substitution should also precede the addition/removal of misspellings, since our synonym substitution will not be capable of substituting misspelled words. Only `_spch` is essentially independent of the other countermeasures and therefore could be placed at any point in the ordering.

In conclusion, we end up with seven different combinations of our countermeasures: `_spell_syn`, `_spell_spch`, `_syn_mis`,`_syn_spch`, `_mis_spch`, `_spell_syn_spch` and `_syn_mis_spch`.

Optimizing Countermeasures. For optimizing countermeasures, we try to make the affected entity more similar to a pre-chosen target entity. We consider two different methods for choosing a suitable target, and introduce them in the following.

Optimizing to the Average: The first method simply chooses the average entity of the community as the target, thus trying to align the current entity's feature distribution with the community's overall feature distribution (cf. Definition 26).

Optimizing using Anonymous Subsets. The second method makes use of the (k, d)-anonymous subsets to find a (possibly) more suitable target entity: intuitively, this method tries to find an (actually existing) entity close by that has

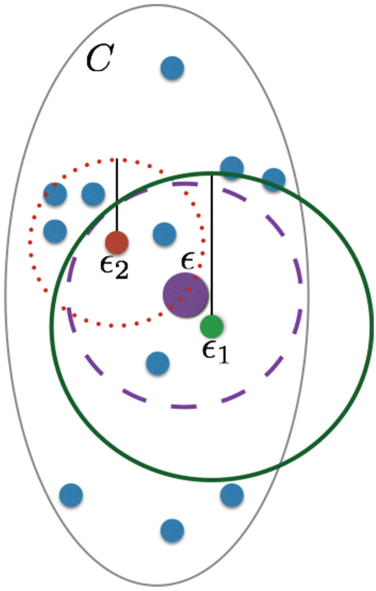


Fig. 5. Optimizing countermeasure using anonymous subsets. (Color figure online)

many other entities within its near environment. Figure 5 illustrates the following, more formal definition of this method: First, we compute the entity ϵ's (k, d)-anonymous subset $\mathcal{A}_\epsilon$ for a given k, such that d is minimal (indicated by the purple, dashed circle). Then, we compute the (k, d')-anonymous subset for every $\epsilon' \in \mathcal{A}_\epsilon$ for the same k and choose that entity ϵ' as the target, which has the smallest d. In our illustration, ϵ_2's anonymous subset (the red, dotted circle) has the smallest convergence and, thus, would be chosen as the countermeasure's target for ϵ.

Given a target ϵ', consider a optimizing countermeasure $\mathcal{C}$ that could replace the current word *house* by its synonym *domicile*. Then, $\mathcal{C}$ would first estimate the similarity to the target for both actions – *keeping house* and *replacing it by domicile* – and choose that action that provides the highest similarity (no matter if ϵ' is the average entity or an actually existing entity like ϵ_2).

7.3 Methodology

On a higher level, our evaluation consists of four parts, which are depicted in Fig. 6:

(1) *A-Priori Weight Determination*: Both, authorship attribution as well as assessing the effectiveness of our countermeasures, require training a classifier to obtain the weights $\boldsymbol{\lambda}$ and the intercept ρ to determine the similarity of entities (cf. Definition 25). Thus, we first determine optimal weights $\lambda_{\mathsf{F1}}, \ldots, \lambda_{\mathsf{F33}}$ for our dataset by extracting the features from each text passage and computing

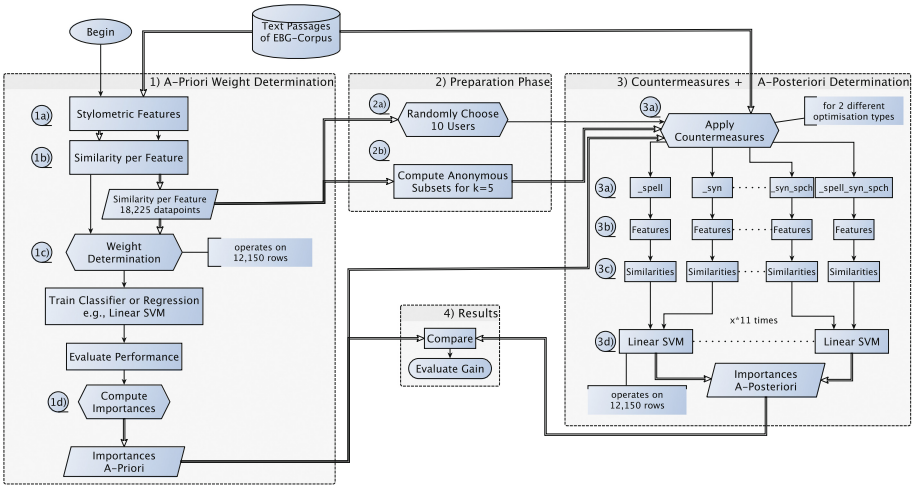


Fig. 6. This flowchart represents our actual set-up and methodology.

the similarity per feature. Ideally, we then compare the weights returned by multiple different optimization techniques to obtain the best performing set of weights. In the context of this paper, however, we simply consider the weights produced by training a linear support vector machine (SVM) using 10-fold cross validation, as exemplified in [3], with the features F1 to F33, since this approach has proven to be well-performing for the task of author-attribution.

(2) *Preparation Phase*: We assume that from the set of all authors, only one actually deploys countermeasures at any given time: this simulates the scenario where the community, together with all related text passages, already exists, and a chosen test author wants to privately publish a new text passage into this community. To capture this scenario, we select a set of test authors for which we evaluate the application of countermeasures before the text passage is published.

For our optimizing countermeasures, we pre-compute the (k, d)-anonymous subsets for the target selection performed by these countermeasures. A detailed description of this part is presented in Sect. 7.3.

(3) *Application of Countermeasures and A-Posteriori Weight Determination*: In this part, we generate the test authors' text-passages after the application of the countermeasures introduced in Sect. 7.2. We then calculate the new feature distributions and the resulting similarities for every countermeasure application to derive the a-posteriori weights $\boldsymbol{\lambda}'$ as well as the intercept ρ'. A detailed description of this part is presented in Sect. 7.3.

(4) *Results*: This last part finally computes the gain for each countermeasure given the a-priori and a-posteriori weights and will be discussed in Sect. 8.

Figure 6 depicts the overall methodology behind our experiments to the point of each single computation step. While the gray boxes represent the structure given above, we also numbered each step for reference purposes.

A-Priori Weight Determination

Step (1a,b). The goal of the first part is to determine the weights $\lambda_{\mathsf{F1}}, \ldots, \lambda_{\mathsf{F33}}$ as well as the intercept ρ as required in Definitions 25 and 28. To this end, we first determine the feature frequencies *(1a)* and compute the resulting similarities per feature *(1b)*.

The similarities are not only computed between entities across communities, but also between entities within the same community. While only the first kind of similarities is needed for the training (resulting in 12,150 pairs of entities) and the authorship attribution, the second kind of similarities is needed for the determination of (k, d)-anonymous subsets within the communities.

Step (1c). Next, we apply regression or classification algorithms to determine the weights $\lambda_{\mathsf{F1}}, \ldots, \lambda_{\mathsf{F33}}$ and the intercept ρ in such a way that matching entities, i.e. entities that belong to the same author, receive a high similarity score, whereas non-matching entities receive a low one.

Conceptually, we could apply various methods, such as simple linear regression, regularized linear regression etc., compare their output and choose the best performing weights. Due to space restrictions, however, we directly choose the classification via linear support vector machines (SVMs) using 10-fold cross validation, since SVMs have already shown promising results in previous work on authorship attribution [3,10]. The resulting weights for each feature are depicted in the fifth column of Table 2 and are directly obtained from the decision function $D(\boldsymbol{x}) = \boldsymbol{\lambda} \cdot \boldsymbol{x} + \rho$ of linear SVMs.

Step (1d): In this final a-priori steps we then take the output of the classifier above and determine the importance (cf. Definition 28) of each feature class in discriminating the different authors' writing styles. We will later compare this a-prior importance value of each feature with their importance determined after applying a countermeasure to determine the countermeasure's gain (cf. Definition 29).

Preparation Phase. The purpose of the preparation phase is to generate a list of test authors and to prepare further data necessary for the countermeasures to be applied.

Step (2a). The set of test authors is randomly chosen from the whole range of available authors in our dataset. We choose a representative set of least 20 % of the available authors as test authors, which resulted in 10 authors for our test set.

Step (2b). For applying optimizing countermeasures using the (k, d)-anonymous subset technique, we also have to compute the anonymous subsets within each community. For this task, we fixed $k = 5$, which corresponds to

Table 3. Total gains and matching accuracy at top k.

Countermeasure	Gain	Top 1	Top 5	Top 10	Top 15	Top 20
Before countermeasures	-	0.9821	0.98613	0.96025	0.93395	0.9065
`_spell_syn`	0.00948	0.98062	0.98062	0.95556	0.92893	0.90189
`_spell_syn_spch`	0.00934	0.98053	0.98029	0.95498	0.92844	0.90156
`_syn`	0.00879	0.98053	0.98119	0.9558	0.92951	0.90247
`_syn_spch`	0.00876	0.98045	0.98095	0.95572	0.92901	0.90214
`_syn_mis`	0.00363	0.98053	0.98078	0.95564	0.92885	0.9023
`_syn_mis_spch`	0.00357	0.98045	0.98053	0.95539	0.92847	0.90206
`_spell_spch`	0.00069	0.982181	0.98638	0.960576	0.93403	0.90658
`_spell`	0.00062	0.98193	0.98564	0.95967	0.93337	0.90593
`_spch`	−0.00042	0.98235	0.98695	0.96107	0.93453	0.90716
`_mis`	−0.0018	0.98202	0.98613	0.96058	0.93362	0.90593
`_mis_spch`	−0.00227	0.98226	0.9863	0.96132	0.93428	0.90658

approximately 10 % of the entities in a community in our dataset. This way, the $(5, d)$-anonymous subset tells us how similar the closest 10 % of the community is at least.

Application of Countermeasures and A-Posteriori Weight Determination. This section deals with the actual application of the countermeasures on our dataset *(3a)* and the a-posteriori weight and importance determination *(3b–3d)*.

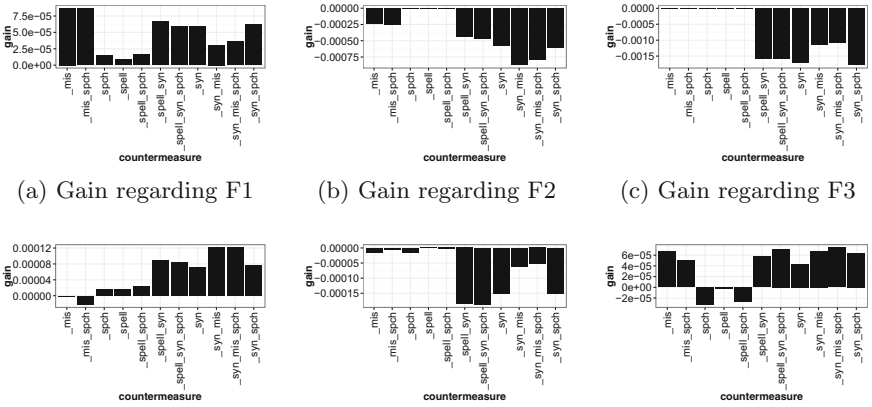
Step (3a). We apply each countermeasure $\mathcal{C}$ separately to the original text-passages m_i of our test authors, yielding modified text-passages $\mathcal{C}(m_i)$ for every countermeasure. Afterwards, we yield further modified messages by following the countermeasure combinations presented in Sect. 7.2, each modified message recorded separately.

Step (3b,c). The next step on the way towards the countermeasures' gains is the computation of the new feature distributions and, thereafter, the corresponding similarities. This time, we only need to compute the similarities across communities, resulting in a total of 12,150 pairs of entities together with their corresponding similarity, forming the base for the following a-posteriori weight determination.

Step (3d). For the a-posteriori weight determination, we follow the same approach as in step *(1c)*: For every application of countermeasures, we compute the optimal weights $\boldsymbol{\lambda}$ and the intercept ρ, and therefrom derive the corresponding importance for each feature class.

8 Evaluation of Countermeasures

Before reviewing the actual results of our countermeasures, we first examine the use of the *term frequency inverse document frequency* (tf-idf) for our statistical models and analyze the impact of the optimization strategy of our countermeasures.



(a) Gain regarding F1 (b) Gain regarding F2 (c) Gain regarding F3

(d) Gain regarding F7 (e) Gain regarding F8 (f) Gain regarding F9

Fig. 7. Gain regarding different feature classes plotted for each countermeasure.

TF-IDF. At least regarding our dataset, the use of the tf-idf for our statistical models does not substantially improve the matching accuracy. When considering only that entity with the highest similarity to a given target entity as matching, the number of true positives increases only by 1 when using tf-idf. When considering the top 15 as matching, the accuracy with the usage of tf-idf is even worse than without. In total, as the use of tf-idf would only increase the complexity of our methodology without providing substantial benefit, we decided to rely on the features' frequencies only and did not consider tf-idf any further in our evaluation.

Optimizing Countermeasures. When comparing both optimization strategies for our countermeasures, optimization to the average and optimization using anonymous subsets, the second one provided the better results for our evaluation. In some cases the optimization to the average results in larger changes to the accuracy (with a maximum change of 0.00634 for `_syn_mis` in the top 20 accuracy), because the artificial target might be very dissimilar to the entity applying the countermeasure and, thus, more likely results in substantial changes to the features.

However, it frequently happens that the averaged entity of a community is not the best target for our countermeasures: Consider a community with only three entities, two of which are far away from each other. Then placing the third entity in the middle of the others yields a higher identifiability compared to placing it beneath one of the others. In the latter case, two entities are nearly indistinguishable, while in the first case all entities are clearly distinguishable. We therefore focus on the optimization using anonymous subsets in our discussion.

8.1 Observations

We now present the results obtained by following the methodology presented in the last section. In Figs. 7 and 8 we illustrate the gain for some of the feature classes individually (cf. Definition 29). A global comparison of all countermeasures and their gains with respect to each feature can be found in Fig. 9 in the Apppendix.

Some of the observed gains are negative: in these cases, the countermeasure caused an increase in the importance of the corresponding feature class. For example, applying countermeasure `_mis` (Misspellings) results in a significant increase of feature F30's (word unigrams) weight, i.e., making it more significant in the authorship recognition task.

The overall gain scored by each countermeasure is illustrated in Table 3: the gains are given in absolute values, summing all feature specific gains. Moreover, we show also the matching accuracy when considering the top k entities regarding their similarity to our target entity as matching. Note that we trained our classifier on the whole data set using 10-fold cross validation. We therefore get a very high accuracy rating, and the countermeasures have a rather low absolute gain overall. We only use the presented values for a relative comparison of the gains achieved by each countermeasure. A practical, absolute assessment would require us to make additional assumptions on how an adversary trains his classifier, and the presented results can be seen as a worst-case estimation at best.

Interestingly, most of our countermeasures have a positive total gain, with the only exceptions being `_mis`, `_spch` and `_mis_spch`. While `_mis` seems to generally replace almost all words by their misspelling and thereby facilitates the matching of those entities, `_spch` performs better, but nevertheless is not optimal in its decisions. In contrast, the best countermeasures are those involving `_syn` and `_spell`: Although `_spell` alone does not seem powerful enough to change a lot (also due to the small amount of spelling mistakes in our dataset), its combination with `_syn` seems to help the synonym replacement, which is able to shift the weights into the desired direction.

We can also see that, in almost all cases, a higher total gain also implies a decreased matching accuracy. Figure 8f depicts this relation exemplary for the top 1 accuracy.

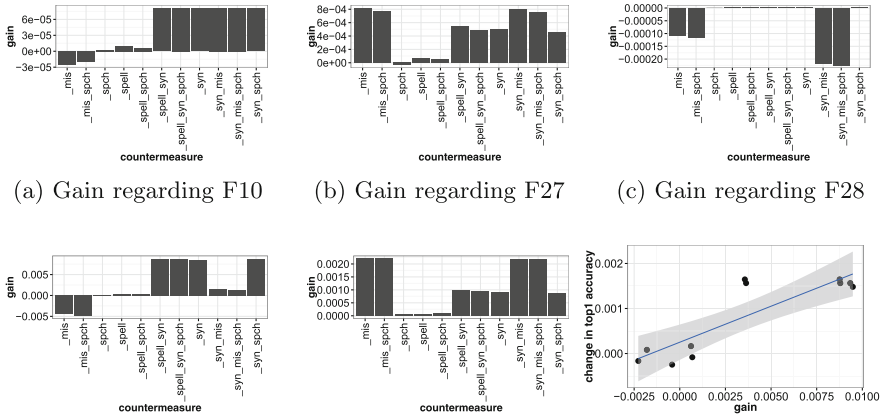
Furthermore, Fig. 9 in the appendix clearly shows that only those weights change for which we expect a modification by our countermeasures (if the weights do not change, the gain is 0 according to our definition). A more detailed and in depth explanation for some of the feature classes will be presented in the next subsection.

8.2 Discussion

We now discuss the results observed in the last section and provide in depth explanations for the gains achieved for the most interesting feature classes. Notice, however, that we use different scales on the y-axes in the Figs. 7 and 8 for better readability. For a more comprehensive comparison of each countermeasure's gain per feature class please refer to Fig. 9.

Letter Unigrams, Bigrams and Trigrams. While all of our countermeasures have a very small positive gain for letter unigrams (F1), this is certainly not true for letter bigrams (F2) and trigrams (F3), which both have negative gains for most countermeasures and especially those involving `_syn`. To further investigate this, we start by looking at the letter bigrams (F2) for the `_syn` countermeasure and trace back the reason for the negative gain:

Letter Bigrams. One frequent action by our synonym replacer is to replace adjectives by participles (e.g., *afraid* → *frightened*), which results in an increased



(a) Gain regarding F10 (b) Gain regarding F27 (c) Gain regarding F28

(d) Gain regarding F30 (e) Gain regarding F31 (f) Total gain compared to change in Top 1 accuracy

Fig. 8. Gain regarding different feature classes plotted for each countermeasure and comparison of total gain.

use of the bigram *ed* for our test authors. In fact, the frequency of *ed* increased by approximately 100 usages for every test author.

Another frequent action by the synonym replacer is caused by the natural language generation tool having problems with some adjectives and adverbs: often, it replaces *most* by *mostest* and thereby increases the use of *es* in a similar magnitude as of *ed*.

Letter Trigrams. Next, we also explore `_syn`'s changes with respect to letter trigrams (F3). Here, the most interesting change is the increased frequency of *ive*, which is caused by frequent replacements of *have* with *give* and forms of *to be* with *live*.

Letter Unigrams. In general, all of the aforementioned changes in fact facilitate the matching of our test authors and thus provide a negative gain. However, although the changes also affect the letter unigrams, we can observe a small positive gain for this feature class. While we especially notice an increased usage of *e*, this letter is frequent in our dataset anyway (and in English in general) and thus does not contribute to a facilitated matching as much as the combinations in letter bigrams and trigrams.

Punctuation and Special Characters. Since the gains are very different among all four feature classes (F7-F10), we directly discuss their results individually. However, it is important to note that the gain of both special character feature classes is nearly zero when compared with others in Fig. 9.

Unicode Punctuation. The unicode punctuation feature class (F7) reveals an interesting phenomenon: while the `_mis` countermeasure provides almost no gain and the `_syn` countermeasure provides a positive gain, the combination of both countermeasures further increases the positive gain.

A careful examination shows that the `_mis` countermeasure only changes this feature very little by introducing misspelled variants with *'* in it, e.g., *countries* → *countrie's*. The `_syn` countermeasure primarily replaces words like *double* with compound words as *two-fold*, changing the feature distribution more substantially. In combination, the application of the `_mis` countermeasure after `_syn` yields much more added *'* than without the combination and, thus, is able to further improve the gain.

Writeprints Punctuation. Regarding the writeprints punctuation feature class (F8), all of our countermeasures provide either a gain close to 0 or a negative gain. While the gain close to 0 can be observed for those feature classes,

which have no real impact on the punctuation, the negative gain clearly is caused by the `_syn` feature class as it is present in all those countermeasures.

The reason for the negative impact of the `_syn` countermeasure is that it introduces new punctuation for our test authors when replacing *a(n)* by *one's*. Since our dataset contains more formal writing, this punctuation character has not been used very frequently (159 times for our test authors) before the countermeasure's application, such that the increased usage (341 times for our test authors) helps in identifying them.

Unicode Special Characters. Interestingly, the gain of our dedicated `_spch` countermeasure is negative regarding the Unicode special characters (F9), while other countermeasures can achieve a positive gain here. When inspecting the reason for that, however, it becomes clear that for example the `_syn` countermeasure does not change the unicode special characters at all and the very small gain is only caused by the SVM. This shows that it is very hard to reason about gains close to 0 and it is better to focus on the substantial gains.

Nevertheless, it is worth noting that our `_spch` countermeasure succeeds in removing special characters from the test author's writing, but thereby facilitates the identification of other authors.

Writeprints Special Characters. Finally, we also take a look at the Writeprints special characters (F10), for which the gains have approximately the same small magnitude compared to the Unicode Special Characters. Again, we can observe the phenomenon that very small gains can be caused by the SVM without changes in the actual features in case of the `_mis` countermeasure.

The most notable, but nevertheless small change in the actual features is due to the `_syn` countermeasure, which increases the frequency of - because of compound words like *two-fold* (cf. Unicode Punctuation).

Hapax Legomena. Hapax Legomena (F27) are of difficult nature, as the same action can increase or decrease their frequency only depending on the surrounding text. Fortunately, both `_mis` and `_syn` countermeasures achieve a positive gain for this feature class.

Our `_mis` countermeasure mainly creates new hapax legomena by replacing all occurrences except for one by a misspelled variant, e.g., for words like *from* ($\rightarrow$ *fomr*), who often appear as hapax legomena in other text passages. Unfortunately, it cannot eliminate hapax legomena, because replacing such a word by a misspelled variant only yields a new, uniquely appearing word.

The `_syn` countermeasure often eliminates hapax legomena by replacing those with compound words whose components are more frequent within the text, e.g., *are* $\rightarrow$ *make up*.

This way, both optimizing countermeasures are able to harden the matching in our dataset, at least concerning the hapax legomena.

Misspelled Words. Since we have a dedicated countermeasure for misspelled words (F28), we also explore the very small, but negative gain caused by our `_mis` countermeasure.

Clearly, misspelled words were nearly unimportant for the matching before the countermeasures' applications (there were only 853 words identified as misspelled in the whole dataset). However, after the application of the `_mis` countermeasure, 13,212 misspellings can be found in the dataset, naturally resulting in a larger importance during the matching.

Word Unigrams and Bigrams. The last two features, which we will examine in more detail, are word unigrams (F30) and word bigrams (F31). Especially word unigrams appear to be the most important feature class in our dataset, so that we will conclude its analysis with possible reasons for the countermeasures total gains.

Word Unigrams. When examining the gains of our countermeasures regarding word unigrams (F30), it becomes visible that `_syn` and `_mis` have the most impact on our test authors. While `_syn` provides a positive gain, mainly by blending into the vocabulary and word frequencies of other authors, `_mis` provides only a negative gain, because it introduces a lot of misspelled words, thereby facilitating the matching. Moreover, as already mentioned for the hapax legomena, the `_mis` countermeasure often replaces all occurrences except for one by a misspelling, which on the one hand influences the hapax legomena in a positive way, but on the other hand has a negative impact on the word unigrams.

Word Bigrams. Similar to word unigrams, here, the `_syn` countermeasure also is able to adapt word bigram (F31) frequencies of our test authors to those that are present in our dataset anyway. Interestingly, the `_mis` countermeasure produces a positive gain for word bigrams, although the gain for word unigrams was negative. While, in contrast to the other feature classes, we did not find a compelling reason for that during our examination, we believe that this happens because of the strong correlation between word unigrams and bigrams: As the importance of word unigrams increases, the importance of word bigrams decreases.

9 Conclusion and Future Work

We presented a user-centric privacy framework for reasoning about privacy in open web settings. In our formalization, we address the essential challenges of

privacy in open settings: we defined a comprehensive data model that can deal with the unstructured dissemination of heterogeneous information, and we derived the sensitivity of information from user-specified and context-sensitive privacy requirements. We showed that, in this formalization of privacy in open settings, hard security guarantees in the sense of Differential Privacy are impossible to achieve. We then instantiated the general framework to reason about the identity disclosure problem. The technical core of our identity disclosure model is the new notion of (k, d)-anonymity that assesses the anonymity of entities based on their similarity to other entities within the same community. We applied this instantiation to a dataset of 15 million user-generated text entries collected from the Online Social Network Reddit and showed that our framework is suited for the assessment of linkablity threats in Online Social Networks.

In a second step, we extended the linkability model we derived from general privacy framework, and provided the foundations for comprehensively assessing the effectiveness of countermeasures against authorship recognition. Central to this formalization is the notion of *gain* with which we quantify how well a countermeasure achieves reduces the significance of identifying writing style features. We evaluate this formalization on the Extended-Brennan-Greenstadt corpus [10,11]. In our evaluation we follow a comprehensive experimental methodology we also introduce in this work, structuring the evaluation process and allowing for an easy extension. We then evaluate four different countermeasures, one simple and three optimizing, and their combinations and discuss the reduction regarding feature importance they achieved.

As far as future work is concerned, many directions are highly promising. First, our general framework only provides a static view on privacy in open settings. Information dissemination on the Internet, however, is, in particular, characterized by its highly dynamic nature. Extending the model presented in this paper with a suitable transition system to capture user actions might lead to powerful system for monitoring privacy risks in dynamically changing, open settings. Second, information presented in Online Social Networks is often highly time-sensitive, e.g., shared information is often only valid for a certain period of time, and personal facts can change over time. Explicitly including timing information in our entity model will hence further increase the accuracy of the entity models derived from empirical evidence. Finally, our privacy model is well-suited for the evaluation of protection mechanisms for very specific privacy requirements, and new such mechanisms with provable guarantees against restricted adversaries can be developed. On the long run, we pursue the vision of providing the formal foundations for comprehensive, trustworthy privacy assessments and, ultimately, for developing user-friendly privacy assessment tools.

A Countermeasure Gain

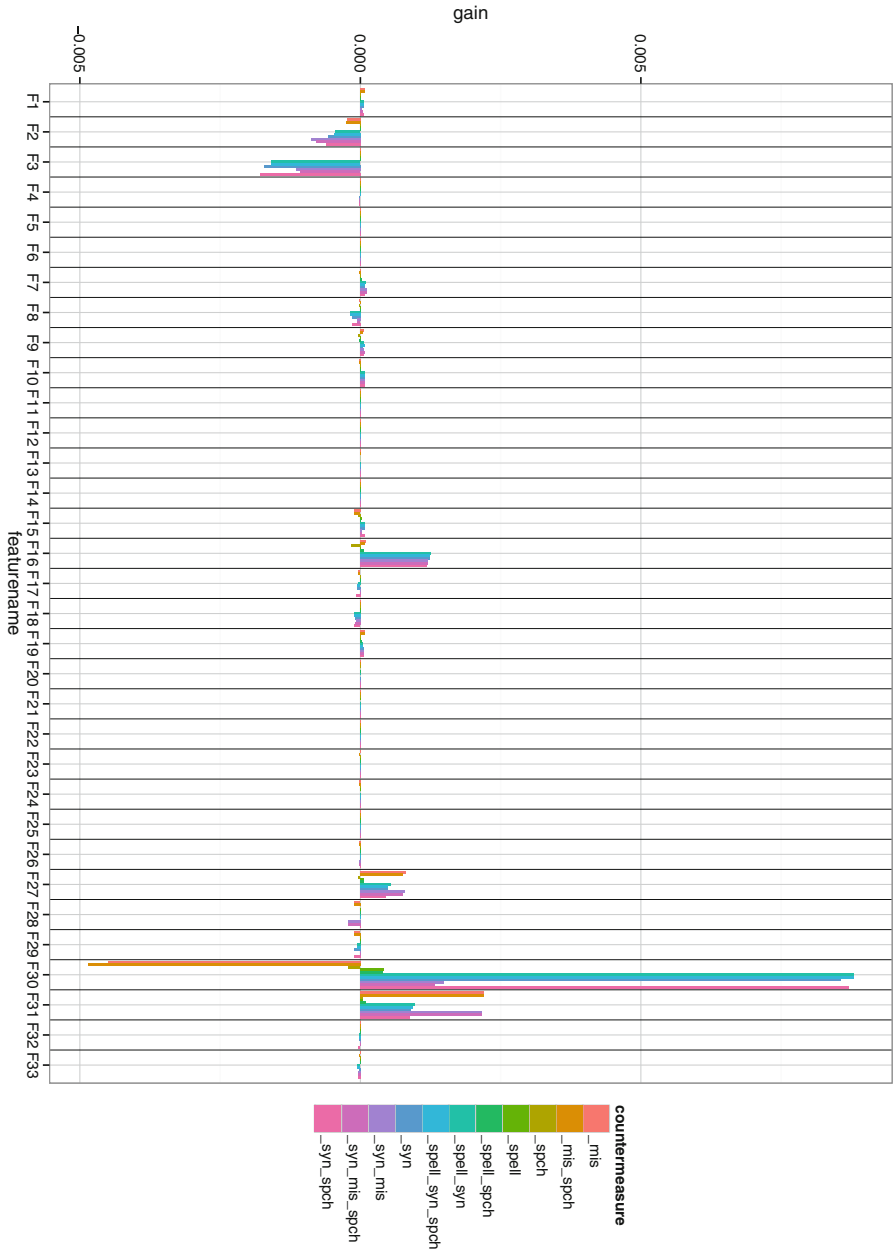


Fig. 9. All gains in a global comparison.

References

1. The online social network reddit. http://www.reddit.com. Accessed Sept 2015
2. Directive 95/46/EC of the European Parliament and of the Council on the Protection of Individuals with Regard to the Processing of Personal Data and on the Free Movement of Such Data (1996)
3. Abbasi, A., Chen, H.: Writeprints: a stylometric approach to identity-level identification and similarity detection in cyberspace. ACM Trans. Inf. Syst. (TOIS) **26**(2), 1–29 (2008)
4. Afroz, S., Brennan, M., Greenstadt, R.: Detecting hoaxes, frauds, and deception in writing style online. In: Proceedings of the 33rd IEEE Symposium on Security and Privacy (S&P), pp. 461–475 (2012)
5. Afroz, S., Islam, A.C., Stolerman, A., Greenstadt, R., McCoy, D.: Doppelgänger finder: taking stylometry to the underground. In: Proceedings of the 35th IEEE Symposium on Security and Privacy(S&P), pp. 212–226 (2014)
6. Anonymouth. https://www.cs.drexel.edu/~pv42/thebiz/
7. Backes, M., Kate, A., Manoharan, P., Meiser, S., Mohammadi, E.: AnoA: a framework for analyzing anonymous communication protocols. In: Proceedings of the 26th IEEE Computer Security Foundations Symposium (CSF), pp. 163–178 (2013)
8. Balduzzi, M., Platzer, C., Holz, T., Kirda, E., Balzarotti, D., Kruegel, C.: Abusing social networks for automated user profiling. In: Jha, S., Sommer, R., Kreibich, C. (eds.) RAID 2010. LNCS, vol. 6307, pp. 422–441. Springer, Heidelberg (2010)
9. Bambauer, J., Muralidhar, K., Sarathy, R.: Fool's gold! An illustrated critique of differential privacy. Vanderbilt J. Entertainment Technol. Law **16**(4), 701–755 (2014)
10. Brennan, M.R., Afroz, S., Greenstadt, R., Stylometry, A.: Circumventing authorship recognition to preserve privacy and anonymity. ACM Trans. Inf. Syst. Secur. (TISSEC) **15**(3), 12:1–12:22 (2012)
11. Brennan, M.R., Greenstadt, R.: Practical attacks against authorship recognition techniques. In: Proceedings of the 21st Annual Conference on Innovative Applications of Artificial Intelligence (IAAI) (2009)
12. Bromby, M.: Security against crime: technologies for detecting and preventing crime. Int. Rev. Law **20**(1–2), 1–6 (2007)
13. Calì, A., Calvanese, D., Colucci, S., Di Noia, T., Donini, F.M.: A logic-based approach for matching user profiles. In: Negoita, M.G., Howlett, R.J., Jain, L.C. (eds.) KES 2004. LNCS (LNAI), vol. 3215, pp. 187–195. Springer, Heidelberg (2004)
14. Chaski, C.E.: Who's at the keyboard? Authorship attribution in digital evidence investigations. Int. J. Digit. Evid. **4**(1), 1–13 (2005)
15. Chatzikokolakis, K., Andrés, M.E., Bordenabe, N.E., Palamidessi, C.: Broadening the scope of differential privacy using metrics. In: De Cristofaro, E., Wright, M. (eds.) PETS 2013. LNCS, vol. 7981, pp. 82–102. Springer, Heidelberg (2013)
16. Chen, R., Fung, B.C.M., Philip, S.Y., Desai, B.C.: Correlated network data publication via differential privacy. VLDB J. **23**(4), 653–676 (2014)
17. Chen, T., Kaafar, M.A., Friedman, A., Boreli, R.: Is more always merrier? A deep dive into online social footprints. In: Proceedings of the 2012 ACM Workshop on Online Social Networks (WOSN), pp. 67–72 (2012)
18. The cmu pronouncing dictionary (version 0.7b). http://www.speech.cs.cmu.edu/cgi-bin/cmudict. Accessed Feb 2015
19. Cortis, K., Scerri, S., Rivera, I., Handschuh, S.: Discovering semantic equivalence of people behind online profiles. In: Proceedings of the 5th International Workshop on Resource Discovery (RED), pp. 104–118 (2012)

20. Derczynski, L., Ritter, A., Clark, S., Bontcheva, K.: Twitter part-of-speech tagging for all: overcoming sparse and noisy data. In: Proceedings of RANLP, pp. 198–206 (2013)
21. Dinur, I., Nissim, K.: Revealing information while preserving privacy. In: Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems (PODS), pp. 202–210 (2003)
22. Dwork, C.: Differential privacy: a survey of results. In: Proceedings of the 5th International Conference on Theory and Applications of Models of Computation, pp. 1–19 (2008)
23. Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., Naor, M.: Our data, ourselves: privacy via distributed noise generation. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 486–503. Springer, Heidelberg (2006)
24. Dwork, C., Naor, M.: On the difficulties of disclosure prevention in statistical databases or the case for differential privacy. J. Priv. Confidentiality **2**(1), 8 (2008)
25. Endres, D.M., Schindelin, J.E.: A new metric for probability distributions. IEEE Trans. Inf. Theor. **49**(7), 1858–1860 (2003)
26. Fast, G.: Syllable counter. http://search.cpan.org/~gregfast/Lingua-EN-Syllable-0.251/Syllable.pm. Accessed Feb 2015
27. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Massachusetts (1998)
28. Goga, O., Lei, H., Parthasarathi, S.H.K., Friedland, G., Sommer, R., Teixeira, R.: Exploiting innocuous activity for correlating users across sites. In: WWW (2013)
29. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Mach. Learn. **46**(1–3), 389–422 (2002)
30. Heatherly, R., Kantarcioglu, M., Thuraisingham, B.: Preventing private information inference attacks on social networks. IEEE Trans. Knowl. Data Eng. **25**(8), 1849–1862 (2013)
31. Holmes, D.I.: The evolution of stylometry in humanities scholarship. Literary Linguist. Comput. **13**(3), 111–117 (1998)
32. Languagetool spell checker. https://languagetool.org. Accessed Feb 2015
33. Juola, P.: Detecting stylistic deception. In: Proceedings of the 2012 EACL Workshop on Computational Approaches to Deception Detection, pp. 91–96 (2012)
34. Kasivisiwanathan, S.P., Smith, A.: On the 'Semantics' of differential privacy: a Bayesian formulation. J. Priv. Confidentiality **6**(1), 1–16 (2014)
35. Kifer, D., Machanavajjhala, A.: No free lunch in data privacy. In: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data, pp. 193–204 (2011)
36. Koppel, M., Schler, J., Argamon, S.: Computational methods in authorship attribution. J. Am. Soc. Inf. Sci. Technol. **60**(1), 9–26 (2009)
37. Kosinski, M., Bachrach, Y., Kohli, P., Stillwell, D., Graepel, T.: Manifestations of user personality in website choice and behaviour on online social networks. Mach. Learn. **95**(3), 357–380 (2014)
38. Krishnamurthy, B., Wills, C.E.: On the leakage of personally identifiable information via online social networks. In: Proceedings of the 2nd ACM Workshop on Online Social Networks (WSON), pp. 7–12 (2009)
39. Li, N., Li, T.: t-closeness: privacy beyond k-anonymity and l-diversity. In: Proceedings of the 23rd International Conference on Data Engineering (ICDE) 2007
40. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: l-diversity: privacy beyond k-anonymity. ACM Trans. Knowl. Discov. Data **1**(1), 3 (2007)

41. McCallister, E., Grance, T., Scarfone, K.A.: Sp 800–122. Guide to Protecting the Confidentiality of Personally Identifiable Information (PII). Technical report (2010)
42. McDonald, A.W.E., Afroz, S., Caliskan, A., Stolerman, A., Greenstadt, R.: Use fewer instances of the letter "i": toward writing style anonymization. In: Fischer-Hübner, S., Wright, M. (eds.) PETS 2012. LNCS, vol. 7384, pp. 299–318. Springer, Heidelberg (2012)
43. Mendenhall, T.C.: The characteristic curves of composition. Science **9**, 237–249 (1887)
44. Miller, G.A.: WordNet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)
45. Almishari, M., Tsudik, G.: Exploring linkability of user reviews. In: Foresti, S., Yung, M., Martinelli, F. (eds.) ESORICS 2012. LNCS, vol. 7459, pp. 307–324. Springer, Heidelberg (2012)
46. Narayanan, A., Paskov, H., Gong, N.Z., Bethencourt, J., Stefanov, E., Shin, E.C.R., Song, D.: On the feasibility of internet-scale author identification. In: Proceedings of the 33rd IEEE Symposium on Security and Privacy (S&P), pp. 300–314 (2012)
47. Narayanan, A., Shmatikov, V.: Myths, fallacies of "Personally Identifiable Information". Commun. ACM **53**(6), 24–26 (2010)
48. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. In: Proceedings of the 30th IEEE Symposium on Security and Privacy (S&P), pp. 173–187 (2009)
49. Oakes, M.P.: Ant colony optimisation for stylometry: the federalist papers. In: Proceedings of the 5th International Conference on Recent Advances in Soft Computing, pp. 86–91 (2004)
50. Pearl, L., Steyvers, M.: Detecting authorship deception: a supervised machine learning approach using author writeprints. Literary Linguist. Comput. **27**(2), 183–196 (2012)
51. Scerri, S., Cortis, K., Rivera, I., Handschuh, S.: Knowledge discovery in distributed social web sharing activities. In: Proceedings of the 3rd International Workshop on Modeling Social Media: Collective Intelligence in Social Media (MSM) (2012)
52. Scerri, S., Gimenez, R., Herman, F., Bourimi, M., Thiel, S.: digital.me-towards an integrated Personal Information Sphere. In: Federated Social Web Summit Europe (2011)
53. Sharma, N.K., Ghosh, S., Benevenuto, F., Ganguly, N., Gummadi, K.: Inferring who-is-who in the twitter social network. In: Proceedings of the 2012 ACM Workshop on Workshop on Online Social Networks (WSON), pp. 55–60 (2012)
54. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **10**(5), 557–570 (2002)
55. Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, pp. 173–180 (2003)
56. Toutanova, K., Manning, C.D.: Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In: Proceedings of the 2000 Joint SIGDAT conference on Empirical Methods in Natural Language Processing and Very Large Corpora, pp. 63–70 (2000)
57. Uzuner, Ö., Katz, B.: A comparative study of language models for book and author recognition. In: Dale, R., Wong, K.-F., Su, J., Kwong, O.Y. (eds.) IJCNLP 2005. LNCS (LNAI), vol. 3651, pp. 969–980. Springer, Heidelberg (2005)

58. Wikipedia. Lists of common misspellings/for machines. http://en.wikipedia.org/w/index.php?title=Wikipedia:Lists_of_common_misspellings/For_machines&oldid=640791958. Accessed Feb 2015
59. Zheleva, E., Getoor, L.: To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In: Proceedings of the 18th International Conference on World Wide Web (WWW), pp. 531–540 (2009)
60. Zheleva, E., Getoor, L.: Privacy in social networks: a survey. In: Aggarwal, C.C. (ed.) Social Network Data Analytics, pp. 277–306. Springer, New York (2011)
61. Zhou, B., Pei, J.: The k-anonymity and l-diversity approaches for privacy preservation in social networks against neighborhood attacks. Knowl. Inf. Syst. **28**(1), 47–77 (2011)