Exploring Linkability of User Reviews

Mishari Almishari and Gene Tsudik

Computer Science Department, University of California, Irvine
{malmisha,gts}@ics.uci.edu

Abstract

Large numbers of people all over the world read and contribute to various review sites. Many contributors are understandably concerned about privacy in general and, specifically, about linkability of their reviews (and accounts) across multiple review sites. In this paper, we study linkability of community-based reviewing and try to answer the question: *to what extent are ”anonymous” reviews linkable, i.e., highly likely authored by the same contributor?* Based on a very large set of reviews from one very popular site (Yelp), we show that a high percentage of ostensibly anonymous reviews can be accurately linked to their authors. This is despite the fact that we use very simple models and equally simple features set. Our study suggests that contributors reliably expose their identities in reviews. This has important implications for cross-referencing accounts between different review sites. Also, techniques used in our study could be adopted by review sites to give contributors feedback about linkability of their reviews.

1 Introduction

In recent years, popularity of various types of review and community-knowledge sites has substantially increased. Prominent examples include Yelp, Tripadvisor, Epinions, Wikipedia, Expedia and Netflix. They attract multitudes of readers and contributors. While the former usually greatly outnumber the latter, contributors can still number in hundreds of thousands for large sites, such as Yelp or Wikipedia. For example, Yelp had more than 39 million visitors and reached 15 million reviews in late 2010 [1]. To motivate contributors to provide more (and more useful/informative) reviews, certain sites even offer rewards [2].

Some review sites are generic (e.g., Epinions) while others are domain-oriented, e.g., Tripadvisor. Large-scale reviewing is not limited to review-oriented sites; in fact, many retail sites encourage customers to review their products. e.g., Amazon and Netflix.

With the surge in popularity of community-based reviewing, more and more people contribute to review sites. At the same time, there has been an increased awareness with regard to personal privacy. Internet and Web privacy is a broad notion with numerous aspects, many of which have been explored by the research community. However, privacy in the context of review sites has not been adequately studied. Although there has been a lot of recent research related to reviewing, its focus has been mainly on extracting and summarizing opinions from reviews [6, 8, 18] as well as determining authenticity of reviews [10, 11, 13].

In the context of community-based reviewing, contributor privacy has several aspects: (1) some review sites do not require accounts (i.e., allow ad hoc reviews) and contributors might be concerned about linkability of their reviews, and (2) many active contributors have accounts on multiple review sites and prefer these accounts not be linkable. The flip side of the privacy problem is faced by review sites themselves: how to address spam-reviews and sybil-accounts?

The goal of this paper is to explore and measure linkability of reviews by investigating how close and related are a person’s reviews. That is, how accurately we can link a set of anonymous reviews to their original author. Our study is based on over $1,000,000$ reviews and $\simeq\ 2,000$ contributors from Yelp. This paper makes the following contributions:

1. We provide a privacy measurement study where we extensively assess and measure reviews' linkability and show that anonymous reviews are accurately de-anonymized in the presence of very simple features. For example, using only alphabetical letter distributions, we can link up to 83% (and 96% with few additional features) of the anonymous reviews to their real authors. We believe that the findings in this study are very important and alarming for reviewers who are concerned about their privacy.

2. We propose several models and improvements that quite accurately link "anonymous" reviews.

Our results have several implications. One of them is the ability to cross-reference contributor accounts between multiple (and similar) review sites. If a person regularly contributes to two similar review sites under different accounts, anyone can easily link them, since many people tend to consistently maintain their traits in writing reviews. This is possibly quite detrimental to personal privacy. Another implication is the ability to correlate reviews ostensibly emanating from different accounts that are produced by the same author. Our approach can thus be very useful in detecting self-reviewing and, more generally, review spam [10] whereby one person contributes from multiple accounts to artificially promote or criticize products or services.

One envisaged application of our technique is to have it integrated into review site software. This way, review authors could obtain feedback indicating the degree of linkability of their reviews. It would then be up to each author to adjust (or not) the writing style and other characteristics.

Organization: Section 2 provides background information about techniques used in our experiments. The sample dataset and study settings are addressed in Section 3. Next, our analysis methodology is presented in Section 4. Section 5 discusses issues stemming from this work. Then, Section 6 overviews related work and Section7 concludes the paper.

2 Background

This section provides some background about statistical tools used in our study. We use two well-known approaches based on: (1) Naïve Bayes Model [12], (2) Kullback-Leibler Divergence Metric [5]. We briefly describe them below.

2.1 Naïve Bayes Model

Naïve Bayes Model (NB) is a probabilistic model based on the eponymous assumption stating that all features/tokens are conditionally independent given the class. Given tokens: $T_1, T_2, ..., T_n$ in document D, we compute conditional probability of a document class C as follows:

$$P(C|D) = P(C|T_1, T_2, ..., T_n) = \frac{P(T_1, T_2, ..., T_n|C)P(C)}{P(T_1, T_2, ..., T_n)}$$
$$= \frac{P(T_1|C)P(T_2|C).....P(T_n|C)P(C)}{P(T_1, T_2, ..., T_n)}$$

Using the Naïve Bayes assumption,

$$P(T_1, T_2, ..., T_n|C) = P(T_1|C)P(T_2|C).....P(T_n|C)$$

To use NB for classification, we return the class value with maximum probability:

$$Class = argmax_C P(C|D) = argmax_C P(C|T_1, T_2, ..., T_n) \quad (1)$$

Since $P(T_1, T_2, ..., T_n)$ is the same for all C values, and assuming $P(C)$ is the same for all class values, the above equation is reduced to:

$$Class = argmax_C P(T_1|C)P(T_2|C).....P(T_n|C)$$

Probabilities are estimated using the Maximum-Likelihood estimator [5] along with Laplace smoothing [14] as follows:

$$P(T_i|C) = \frac{Num\ of\ T_i\ in\ D\ +\ 1}{Num\ of\ Tokens\ in\ D\ +\ Num\ Possible\ Token\ Values}$$

2.2 Kullback-Leibler Divergence Metric

Kullback-Leibler Divergence (KLD) metric measures the distance between two distributions. For any two distributions P and Q, it is defined as:

$$D_{kl}(P\|Q) = \sum_i P(i)log(\frac{P(i)}{Q(i)})$$

KLD is always positive: the closer to zero, the closer Q is to P. It is an asymmetrical metric, i.e., $D_{kl}(P\|Q) \neq D_{kl}(Q\|P)$. To transform it into a symmetrical metric, we use the following formula (that has been used in [20]):

$$SymD_{kl}(P,Q) = 0.5 \times (D_{kl}(P\|Q) + D_{kl}(Q\|P)) \tag{2}$$

Basically, $SymD_{kl}$ is a symmetrical version of D_{kl} that measures the distance between two distributions. As discussed below, it is used heavily in our study. In the rest of the paper, the term ”KLD” stands for $SymD_{kl}$[1].

3 Data Set and Study Settings

Data Set. Clearly, a very large set of reviews authored by a large number of contributors is necessary in order to perform a meaningful study. To this end, we collected $1,076,850$ reviews for $1,997$ contributors from `yelp.com`, a very popular site with many prolific contributors. The minimum number of reviews per contributor is 330, the maximum – $3,387$ and the average – 539 reviews, with a standard deviation of 354. For the purpose of this study, we limited authorship to prolific contributors, since this provides more useful information for the purpose of review linkage. Note that 50% of the contributors authored fewer than 500 reviews and 76% authored fewer than 600. Only 6% of the contributors exceed $1,000$ reviews. Additionally, 50% of the contributors write reviews shorter than 140 words (on average) and 75% – have average review size smaller than 185. Also, 97% of contributors write reviews shorter than 300 words. The overall average review size is relatively small – 149 words.

Study Settings. Our central goal is to study linkability of relatively prolific reviewers. Specifically, we want to understand – for a given prolific author – to what extent some of his/her reviews relate to, or resemble, others. To achieve that, we first randomly order the reviews of each contributor. Then, for each contributor U with N_U reviews, we split the randomly ordered reviews into two sets:

1. First $N_U - X$ reviews: We refer to this as the **identified record** (IR) of U.
2. Last X reviews: These reviews represent the full set of anonymous reviews of U from which we derive several subsets of various sizes. We refer to each of these subset as an **anonymous record** (AR) of U. An AR of size i consists of the first i reviews of the full set of anonymous reviews of U. We vary the AR size for the purpose of studying the user reviews linkability under different numbers of anonymous reviews.

[1] Note that, under certain conditions, NB and asymmetrical KLD models could be equivalent. That is, $argmax_{Class}P(Class|T_1,T_2,...,T_n)$ is equivalent to $argmin_{Class}D_{kl}(Token\_distribution\|Class\_distribution)$, where $T_1,T_2,...T_n$ are the tokens of a document D and $Token\_distribution$ is their derived distribution. The proof for this equivalency is in [20]. However, this equivalence does not hold when we use the symmetrical version $SymD_{kl}$.

Since we want to restrict the AR size to a small portion of the complete user reviews set, we restrict X to 60 as this represents less than 20% of the minimum number of reviews for authors in our set (330 total). We use the **identified records** (IRs) of all contributors as the training set upon which we build models for linking anonymous reviews. (Note that the IR size is not the same for all contributors, while the AR size is uniform.) Thus, our problem is reduced to matching an anonymous record to its corresponding IR. Specifically, one anonymous record serves as an input to a matching/linking model and the output is a sorted list of all possible account-ids (i.e., IR sets) listed in a descending order of probability, i.e., the top-ranked account-id corresponds to the contributor whose IR represents the most probable match for the input anonymous record. Then, if the correct account-id of the actual author is among top T entries, the matching/linking model has a hit; otherwise, it is a miss. Consequently, our study boils down to exploring matching/linking models that maximize the hit ratio of the anonymous records for varying values of both T and AR sizes. We consider two values of T: 1 (perfect hit) and 10 (near-hit). Whereas, for the AR size, we experiment with a wider range of values which includes: 1, 5, 10, 20, 30, 40, 50 and 60. Note that in this paper, linkability ratio and hit ratio are used exchangeably.

Even though our focus is on the linkability of prolific users, we also attempt to assess performance of our models for non-prolific users. For that, we slightly change the problem setting by making the IR size smaller; this is discussed in Section 4.4.

4 Analysis

As mentioned in Section 2, we use Naïve Bayes (NB) and Kullback-Leibler Divergence (KLD) models. Before analyzing the collected data, we tokenize all reviews and extract four types of tokens:

1. **Unigrams:** set of all single letters. We discard all non-alphabetical characters.

2. **Digrams:** set of all consecutive letter-pairs. We discard all non-alphabetical characters.

3. **Rating:** rating associated with the review. (In Yelp, this ranges between 1 and 5).

4. **Category:** category associated with the place/service being reviewed. There are 28 categories in our dataset,

Note that we experimented our models on larger token sets, namely trigram and stemmed-word sets. Surprisingly, they mostly perform worse(in terms of linkability) than unigrams or digrams. Before proceeding, we re-cap abbreviations and notation in Table 1.

4.1 Methodology

We begin with the brief description of the methodology for the two models.

4.1.1 Naïve Bayes (NB) Model

For each account IR, we built an NB model, $P(token_i|IR)$, from its identified record. Probabilities are estimated using the Maximum-Likelihood estimator [5] and Laplace smoothing [14] as shown in 2. We then construct four models corresponding to the four aforementioned token types. That is, for each IR, we have $P_{unigram}$, P_{digram}, $P_{category}$ and P_{rating}.

To link an anonymous record AR to an account IR with respect to token type R, we first extract all R-type tokens from AR, $T_{R_1}, T_{R_2}, ...T_{R_n}$ (Where T_{R_i} is the i-th R token in AR). Then, for each IR, we compute the probability $P_R(IR|T_{R_1}, T_{R_2}, ...T_{R_n})$. Finally, we return a list of accounts sorted in decreasing order of probabilities. The top entry represents the most probable match.

NB	Naïve Bayes Model
KLD	Symmetrical Kullback-Leibler Model
R	Token Type: rating, unigram or digram
LR	Linkability Ratio
AR	Anonymous Record
IR	Identified Record
$SymD_{KLD}(IR, AR)$	symmetric KLD between IR and AR
$SymD_{KLD\_r}$	symmetric KLD of rating tokens
$SymD_{KLD\_c}$	symmetric KLD of category tokens
$SymD_{KLD\_l}$	symmetric KLD of lexical tokens
$SymD_{KLD\_r\_c}$	symmetric KLD of rating and category
$SymD_{KLD\_l\_r\_c}$	symmetric KLD of all tokens

Table 1: Notation and abbreviations.

4.1.2 Kullback-Leibler Divergence (KLD) Model

We use symmetric KLD (see Section 2) to compute the distance between anonymous and identified records. To do so, we first compute distributions of all records and then we smooth the distributions via Laplace smoothing [14](same as the probability estimation in explained in Naive Bayesian in Section 2). As before, we compute four distributions. To link AR with respect to token type R, we compute $SymD_{kl}$ between the distribution of R for AR and the distribution of R for each IR. Then, we return a list sorted in ascending order of $SymD_{KLD}(IR, AR)$ values. The first entry represents the account with the most likely match.

4.2 Study Results

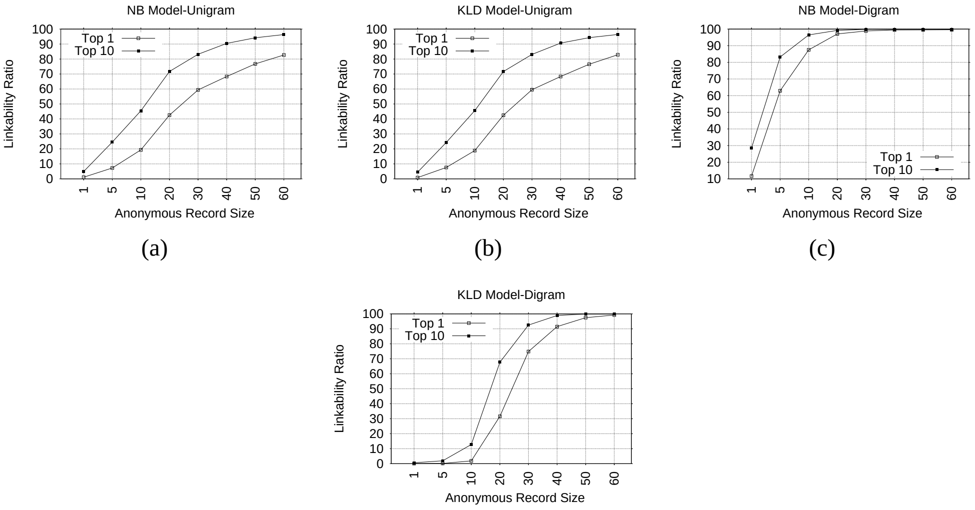
We now present the results corresponding to the lexical tokens. Then, in the next section, we experiment with some combinations of lexical and non-lexical ones.

4.2.1 Lexical – Results

Figures 1(a) and 1(b) depict LRs (Top-1 and Top-10) for NB and KLD with the unigram token. As expected, with the increase in the anonymous record size, the LR grows: it is high in both Top-1 and Top-10 plots. For example, in Top-1 of both figures, the LRs are around: 19%, 59% and 83% for anonymous record sizes of 10, 30 and 60, respectively. Whereas, in Top-10 of both figures, the LRs are around: 45.5%, 83% and 96% for same record sizes. This suggests that reviews are highly linkable based on trivial single-letter distributions. Note that the two models exhibit similar performance.

Figures 1(c) and 1(d) consider the digram token. In both models, the LR is impressively high: it gets as high as 99.6%/99.2% in Top-1 for NB/KLD for an AR size of 60. For example, the Top-1 LRs in NB are: 11.7%, 62.9%, 87.5% and 97.1%, for respective AR sizes of 1, 5, 10 and 20. Whereas, in KLD, the Top-1 LRs for record sizes of 10, 30 and 60 are: 1.9%,74.9% and 99.2%, respectively.

Unlike unigrams – where LRs in both models are comparable – KLD in digram starts with LRs considerably lower than those of NB. However, the situation changes when the record size reaches 50, with KLD performing comparable to NB. One reason for that could be that KLD improves when the distribution of ARs is more similar to that of corresponding identified records; this usually occurs for large record sizes, as there are more tokens.



(a)

(b)

(c)

(d)

Figure 1: LRs of NB and KLD models for unigrams and digrams

Not surprisingly larger AR sizes entail higher LRs. With NB, a larger record size implies that, a given AR has more tokens in common with the corresponding IR. Thus, an increase in the prediction probability $P(IR|T_1, T_2, ...T_n)$. For KLD, a larger record size causes the distribution derived from the AR to be more similar to the one derived from the corresponding IR.

4.3 Improvement I: Combining Lexical with non-lexical Tokens

In an attempt to improve the LR, we now combine the lexical tokens with the non-lexical ones.

4.3.1 Combining Tokens Methodology

This is straightforward in the NB. We simply increase the list of tokens in the unigram- or digram-based NB by adding the non-lexical tokens. Thus, for every AR, we have $P(lexical\_token_i|IR)$, $P(category\_token_i|IR)$ and $P(rate\_token_i|IR)$.

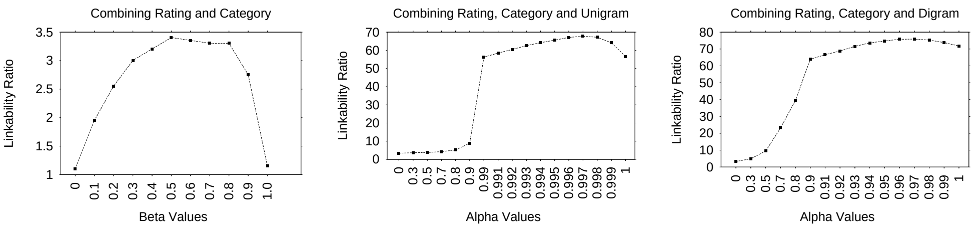
Combining non-lexical with lexical tokens in KLD is less clear. One way is to simply average $SymD_{KLD}$ values for both token types. However, this might degrade the performance, since lexical distributions may convey much more information than their non-lexical counterparts. Thus, giving them the same weight would not yield better results. Instead, we combine them using a weighted average. First, we compute the weighted average of rating and category $SymD_{KLD}$:

$$\begin{aligned} &SymD_{KLD\_r\_c}(P, Q) = \\ &\beta \times SymD_{KLD\_r}(P, Q) + (1 - \beta) \times SymD_{KLD\_c}(P, Q) \end{aligned}$$

Then, we combine the above with $SymD_{KLD}$ of the lexical tokens to compute the final weighted average:

$$\begin{aligned} &SymD_{KLD\_l\_r\_c}(P, Q) = \\ &\alpha \times SymD_{KLD\_l}(P, Q) + (1 - \alpha) \times SymD_{KLD\_r\_c}(P, Q) \end{aligned}$$

Thus, our goal is to get the right β and α values. Intuitively, lexical $SymD_{KLD}$ should have more weight as it carries more information. Since there is no clear way of assigning weight values, we experiment with several choices and pick the one with the best performance; we discuss the selection process below. We experiment only within the IR set and then verify the results generalize to the AR. This is done as follows:



(a) (b) (c)

Figure 2: Results of combining different tokens using different β and α values

First, for every IR, we allocate the last 30 reviews as a testing record and the remainder – as a training record. Then, we experiment with $SymD_{KLD\_r\_c}$ using several β values and set β to the value that yields the highest LR based on the testing records. Then, we experiment with $SymD_{KLD\_l\_r\_c}$ using several α values and, similarly, pick the one with the highest LR.

Since β or α could assume any values, we need to restrict their choices. For β, we experiment with a range of values, from 0 to 1.00 in 0.1 increments. For α, we expect the optimal value to exceed 0.9, since the LR for lexical tokens is probably higher than non-lexical ones. Therefore, we experiment with the weighted average by varying α between 0.9 and 0.99 in 0.01 increments.

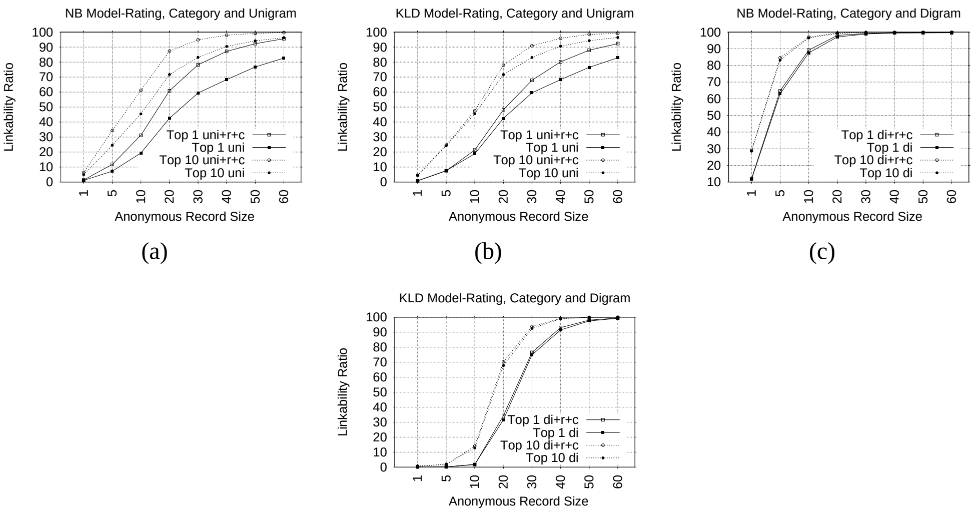
If the values exhibit an increasing trend (i.e., $SymD_{KLD\_l\_r\_c}$ at α of 0.99 is the largest in this range) we continue experimenting in the $0.99 - -1.00$ range in 0.001 increments. Otherwise, we stop. For further verification, we also experiment with smaller α values: $0.0, 0.3, 0.5, 0.7,$ and 0.8, all of which yield LRs significantly lower than 0.9 for both the unigram and digram. We acknowledge that we may be missing α or β values that could further optimize $SymD_{KLD\_l\_r\_c}$. However, results in the following section show that our selection yields good results.

Figure 2(a) shows LRs (Top-1) for β values. The LR gradually increases until it tops off at 3.4% with $\beta = 0.5$ and then it gradually decreases. Figure 2(b) shows LRs (Top-1) for α values in the unigram case. The LR has an increasing trend until it reaches 67.8% with $\alpha = 0.997$ and then it decreases. Figure 2(c) shows LRs (Top-1) for α values in the diagram case where it tops off at 75.9% with $\alpha = 0.97$. Thus, the final values are 0.5 for β and $0.997/0.97$ for $alpha$ in the unigram/digram case. Even though we extract α and β values by testing on a record size of 30, the results in following sections show that the derived weights are effective when tested on ARs of other sizes.

4.3.2 Combining Lexical with Non-Lexical Tokens – Results

Figures 3(a) and 3(b) show Top-1 and Top-10 plots in NB and KLD models of unigram tokens before and after combining them with rating and category tokens. Adding non-lexical tokens to unigrams substantially increases LRs in several record sizes. In NB, the gain in Top-1 LRs ranges from 0.25-18.9% (1.4 - 15.7% for Top-10 LRs). In KLD, the gain in Top-1 LRs ranges from 2.5-11.9 (2-7.8% in Top-10 LRs) for most record sizes. These findings shows how effective is combining the non-lexical tokens with the unigrams. In fact, we can accurately identify almost all ARs.

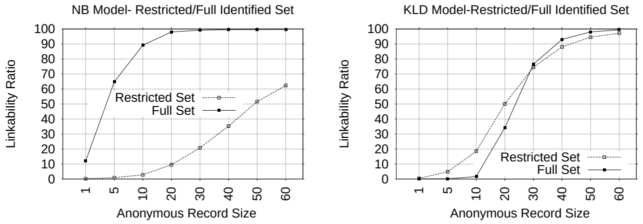
Figures 3(c) and 3(d) show the effect of adding ratings and categories to digrams. The overall effect is less: in NB (KLD) model, the increase in Top-1 LRs ranges from 0.3-1.8% (0.2-2.7%) for most record sizes. The increase is very similar in Top-10 plots.



(a) (b) (c)

(d)

Figure 3: LRs for NB and KLD for combining ratings and categories with unigrams or digrams



(a) (b)

Figure 4: LRs for NB and KLD in full and restricted identified set

4.4 Restricting Identified Record Size

In previous sections, our analysis was based on using the full data set. That is, except for the anonymous part of the data set, we use all of the user reviews as part of our identified set. Although the LR is high in many cases, it is not clear how the models will perform when we restrict the IR size. To this end, we re-evaluate the models with the same problem settings, however, with a restricted IR size. We restrict the IR size to the AR size; both randomly selected without replacement.

Figures 4(a) and 4(b) show two Top-1 plots in NB and KLD models: one plot corresponds to the restricted identified set and the other – to the full set. Tokens used in the models consist of digrams, ratings and categories (since this combination gives the highest LR). Unlike the previous sections, where NB and KLD behaved similarly, the two models now behave differently when restricting the identified set. While NB performs better than KLD on the full set, the latter performs much better than NB when the identified set is restricted. In fact, in some cases, KLD performs better when the set is restricted.

The reason for this improved KLD performance might be the following: in the symmetric KLD distance function, the distributions of both the IR and AR have to be very close in order to match regardless of the size of the IR; unlike the NB, where larger training sets would lead to better estimates of the token probabilities and thus more accurate predictions.

In KLD, we achieve high LRs for many record sizes. For example, Top-1 LRs in the restricted set are 74.5%, 88% and 97.1% when the anonymous (and identified) record sizes are 30, 40 and 60, respectively.

```
Algorithm $Match\_All$: Pseudo Code
Input:  (1) Set of ARs: $S_{AR} = \{AR_1, AR_2, ..., AR_n\}$
        (2) Set of reviewer-ids / identified records:
        $S_{IR} = \{IR_1, IR_2, ..., IR_n\}$
        (3) Set of matching lists for each AR:
        $S_L = \{List_{AR_1}, .., List_{AR_n}\}$
Output: Matching list: $S_M = \{(IR_{i_1}, AR_{j_1}), ..., (IR_{i_n}, AR_{j_n})\}$
set $S_M = \emptyset$
While $|S_{AR}| \neq 0$:
    Find $AR_i$ with smallest $SymD_{KLD}$ in all lists in $S_L$
    Get corresponding reviewer-id $IR_j$
    Add $(IR_j, AR_i)$ to $S_M$
    Delete $AR_i$ from $S_{AR}$
    Delete $List_{AR_i}$ from $S_L$
    For each $List_t$ in $S_L$,
        Delete tuple containing $IR_j$ from $List_t$
    End For
End While
```

NOTE 1: $List_{AR_i}$ in S_L is a list of pairs (IR_j, V_{ij}) where $V_{ij} = SymD_{KLD}(IR_j, AR_i)$, for all j

NOTE 2: $List_{AR_i}$ is sorted in increasing order of V_{ij}, i.e., IR_j with lowest $SymD_{KLD}(IR_j, AR_i)$ at the top.

Figure 5: Pseudo-Code for matching all ARs at once.

Whereas, the LRs in the full set for the same AR sizes are: 76.5% , 93% and 99.4%. When the record size is less than 30, KLD performs better in the restricted set than the full one. For example, when the AR size is 20, the LR in the restricted set is 50.1% and 34.3% in the full set. In NB, Top-1 LR in the restricted set is lower than the full set. For instance, it is 20.8%, 35.3% and 62.4% for AR sizes of: 30, 40 and 60, respectively. Whereas, for the same sizes, the LR is more than 99% in the full set.

This result has one very important implication: even with very small IR sizes, many anonymous users can be identified. For example, with only IR and AR sizes of only 30, most users can be accurately linked (75% in Top-1 and 90% in Top-10). This situation is very common since many real-world users generate 30 or more reviews over multiple sites. Therefore, even reviews from less prolific accounts can be accurately linked.

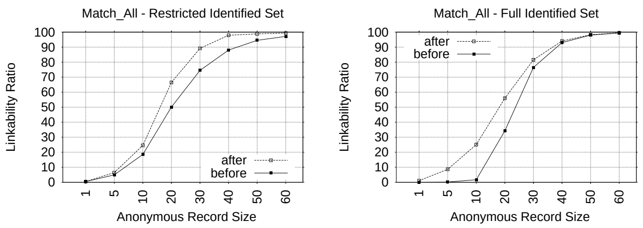
4.5 Improvement II: Matching all ARs at Once

We now experiment with another natural strategy of attempting to match all ARs at once.

4.5.1 Methodology

In previous sections, we focus on independently linking one AR at a time. That is, the input to our matching/linking model is one AR and the output is the user of the closest IR. If we change the problem settings and make the input a set of ARs(instead of one) where each AR belongs to a different user, we may be able to improve the linkabilty knowing that an AR cannot be mapped to more than one user. To this end, we construct algorithm $Match\_All()$ in Figure 5 as an add-on to the KLD models suggested in previous sections where the input is a set of ARs, each of which belongs to a different user. The number of ARs in the input is equal to the number of users in our dataset.

$SymD_{KLD}(IR_j, AR_i)$ symmetrically measures the distance between their (IR_j's and AR_i's) distributions. Since every AR maps to a distinct IR (AR_i maps to IR_i), it would seem that lower $SymD_{KLD}$ would lead to a better match. We use this intuition to design $Match\_All()$. As shown in the figure, $Match\_All()$



(a) (b)

Figure 6: Effects of $Match\_All()$ on LRs in full and restricted identified set: before and after plots

picks the smallest $SymD_{KLD}(IR_j, AR_i)$ as the map between IR_j and AR_i and then deletes the pair (IR_j, V_{kj}) from all remaining lists in S_L. The process continues until we compute all matches. Note that, for any $List_{AR_k}$, (IR_j, V_{kj}) is deleted from the list only when there is another pair (IR_j, V_{lj}) in $List_{AR_l}$, such that $SymD_{KLD}(IR_j, AR_l) \leq SymD_{KLD}(IR_j, AR_k)$, and IR_j has been selected as the match for AR_l.The output of the algorithm is a match-list: $S_M = \{(IR_{i_1}, AR_{j_1}), ..., (IR_{i_n}, AR_{j_n})\}$.

We now consider how $Match\_All()$ could improve the LR. Suppose that we have two ARs: AR_i and AR_j along with corresponding sorted lists L_i and L_j and assume that IR_i is at the top of each list. Using only KLD (as in previous sections), we would return IR_i for both ARs and thus miss one of the two. Whereas, $Match\_All$, would assign IR_i to **only** one AR – the one with the smaller $SymD_{KLD}(IR_i, ...)$ value. We would intuitively suspect that $SymD_{KLD}(IR_i, AR_i) < SymD_{KLD}(IR_i, AR_j)$ since IR_i is the right match for AR_i and thus their distributions would probably be very close. If this is the case, $Match\_All$ would delete IR_i (erroneous match) from the top of L_j which could help clearing up the way for IR_j (correct match) to the top of L_j.

We note that there is no guarantee that $Match\_All()$ will always work: one mistake in early rounds would lead to others in later rounds. We believe that $Match\_All()$ works better if $SymD_{KLD}(IR_i, AR_i) < SymD_{KLD}(IR_j, AR_i)$ ($j \neq i$) holds most of the time.

In the next section, we show the results of $Match\_All()$ when we experiment with the KLD model with digram, rating and category tokens.[2].

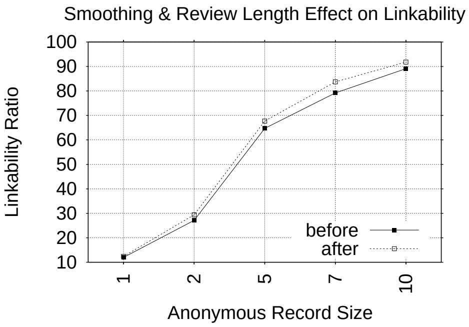
4.5.2 Results

Figures 6(a) and 6(b) show the effect of $Match\_All()$ on Top-1 LRs in both the restricted identified set and the full identified set, respectively. The combination of diagram, rating and category tokens are used. Each figure shows two Top-1 plots: one for the LR after using $Match\_All$ and the other – for the LR before using it. Clearly, $Match\_All$ is effective in improving the LR for almost all record sizes. For the restricted set, the gain in the LR ranges from 1.6-16.4% for nearly all AR sizes. A Similar increase is observed in the full set that ranges from 1-23.4% for most record sizes. This shows that the $Match\_All$ is very effective when used with diagram, rating and category tokens. The privacy implication of $Match\_All$ is important as it significantly increases the LR for small ARs in the restricted set. This shows that privacy of less prolific users is exposed even more with $Match\_All$.

4.6 Improvement III: Improving Linkability for Small Anonymous Records

Although most of the proposed exhibits high LR's when the AR size is large, the linkability is not as high for small record sizes. For improving the LR for small AR (in the full identified set), we consider the NB model that uses diagrams, ratings and categories as its tokens (see Section 4.3.2) as a base for our

[2] We also tried $Match\_All()$ with the NB model and it did not improve the LR.



(a)

Figure 7: Effects of smoothing and review length on LRs: before and after plots

improvement. We use this model as it performs the best for small ARs comparable to other models. To that end, we first change the way we smooth the probabilities as follows:

$$P(token_i|IR) = \frac{Num\ Token_i\ in\ D\ +\ \eta}{Num\ Tokens\ in\ D\ +\ \eta\ \times\ Num\ Possible\ Tokens}$$

Unlike the models in the previous sections (see Section 2)[3], η could take values other than 1. In fact, we experiment with several different values and we find that η value of 0.5 gives the best performance[4]. The intuition is that setting η to a value less than 1 may help downscale the effect of noisy digrams that the user rarely use. Additionally, we leverage the length of the reviews, the number of the alphabetical letters, as an additional feature to the model. We consider the length of the reviews as we intuitively believe that different users tend to write longer/shorter reviews than others. We model the length as a normal distribution and we use the maximum likelihood estimate to set the distribution parameters [14, 5].

Figure 7 shows the effect of this improvement. For clarity, we only show the improvement resulting from combining the two aforementioned steps. As shown, the Top-1 LRs gain roughly ranges from 0.5%-5%. For example, for AR size of 5, 7 and 10, the Top-1 LR approximately increases from 65%, 79% and 89% to 68%, 84% and 92%, respectively. Similar increases are observed in the Top-10 LR which reach 88%/98% for AR size of 5/10 (and up to 30%/54% for AR size of 1/2).

4.7 Study Summary

We now summarize the main findings and conclusions of our study.

1. The LR becomes very high – reaching up to $\sim$ 99.5% in both KLD and NB when using only digram tokens. (See Section 4.2.1).

2. Surprisingly, using only unigrams, we can link up to 83% in both NB and KLD models, with 96% in Top-10. (See Section 4.2.1). This suggests that reviewers expose a great deal merely from their single letter distributions.

3. Non-lexical tokens are very useful in tandem with lexical tokens, especially, the unigram: we observe a $\sim$19%/12% Top-1 LR increase in NB/KLD for some cases. (See Section 4.3.2).

[3] Here, document D refers to the Identified Record IR

[4] Note that we experiment η on only the training set and pick the best value

4. Relying only on unigram, rating and category tokens, we can accurately link 96%/92% of the ARs (size 60) in NB/KLD. (See Section 4.3.2).

5. Restricting the IR size does not always degrade linkability. In KLD, we can link as many as 97% ARs when the IR size is small. (See Section 4.4).

6. Linking all ARs at once (instead of each independently) helps improve accuracy. The gain is up to 16/23% in restricted/full set. (See Section 4.5.2).

7. Generally, NB performs better than KLD when we use the full identified set and KLD performs better when we use the restricted identified set.

8. Combining review length with different smoothing techniques is helpful in increasing the linkability for small AR and the Top-1/Top-10 LR reach 92%/98% for AR size of 10(See Section 4.6).

5 Discussion

Implications. We believe that the results of, and techniques used in, this study have several implications. One implication is the possibility to cross-reference accounts (and reviews) among multiple (similar)review sites. If a person contributes to two similar review sites under two identities, it is likely that sets of reviews from these sites can be linked. This could be quite detrimental to contributors' privacy. Another implication is the ability to correlate – on the same review site – multiple accounts that are in fact manipulated by the same person. This could make our techniques very useful in detecting review spam [10], whereby a contributor authors reviews under different accounts to tout (also self-promote) or criticize a product or a service.

Prolific Users. While there are clearly many more occasional (non-prolific) reviewers than prolific ones, we believe that our study of prolific reviewers is important, for two reasons. First, the number of prolific contributors is still quite large. For example, from only one review site – Yelp – we identified $\sim 2,000$ such reviewers. Second, given the spike of popularity of review sites [1], we believe that, in the near future, the number of such prolific contributors will grow substantially. Also, even many occasional reviewers, with the passage of time, will enter the ranks of "prolific" ones, i.e., by slowly accumulating a sufficient corpus of reviews over the years. Nevertheless, our study suggests that privacy is not high even for non-prolific users, as discussed in Section 4.5. For example, when both IR and AR sizes are only 20 (i.e., total per user contribution is 40 reviews), we can accurately link around 70% of anonymous records to their reviewers.

Anonymous Record Size. Our models perform best when the AR size is 60. However, for every reviewer in our dataset, 60 represents less than 20% of that person's total number of reviews. Also, using NB coupled with digram, rating, category and length features, we can accurately link most anonymous records when AR size is small (see Section 4.6).

Unigram Tokens. While our best-performing models are based on digram tokens, we also obtain high linkability results from unigram tokens that reach up to 83% (96% in the Top 10) in NB or KLD. The results improve to 96/92% when we combine unigrams with rating and category tokens. Note that the number of tokens in unigram-based models is 59 (26) tokens with (without) combining them with rating and category tokens. Whereas, the number of tokens in diagram-based models is 676 (709 when combined with rating and category tokens). This makes linkability accuracy based on unigram models very comparable to its diagram counterpart, while the number of tokens is significantly fewer. This implies a substantial reduction in resources and processing power in unigram-based models which would make them scale better. For example, if we assume that the attacker wants to link a set of anonymous reviews to *many* large review datasets, unigram-based models would scale better, while maintaining similar level of accuracy.

Potential Countermeasures. One concrete application of our techniques is via integration with the review site's front-end software in order to provide feedback to authors indicating the degree of linkability of their reviews. For example, when the reviewer logs in, a linkability nominal/categorical value (e.g. high, medium, and low) could be shown indicating how some of his/her reviews (selected randomly) are linkable to the rest. It would then be up to to the individual to maintain or modify their reviewing patterns to be less linkable. Another way of countering linkability, as suggested in [15], is for the front-end software to automatically suggest a different choice of words that are less revealing (less personal) and more common among many users. We suspect that, with the use of such words, reviews would be less linkable and lexical distributions for different users would be more similar.

6 Related Work

Many authorship analysis studies are in the literature. Among the most related recent studies are [16, 15, 3]. In [16], a large scale author identification techniques (based on linguistic stylometry) are evaluated on blog de-anonymization. While the problem formulation is similar to ours, there are notable differences. First, we study the linkability in a different context; i.e., user reviews. User reviews have ratings and categories, which prove useful in some scenarios, while blogs(used in [16]) do not. Additionally, user reviews are shorter while blogs could be as long as an article. Moreover, user reviews are mainly about user evaluations of a specific service/product while blogs could be very random, such as news reporting or literature-related work. Second, our study points to high linkability ratios in user reviews, nearly 100% Top-1 linkability ratio, where as in [16], the Top-1 linkability ratio is around 20% [5]. Third, our study shows high linkability ratios in the presence of very simple features. A related problem is explored in [15]. It focuses on identifying authors based on reviews in both single- and double-blinded academic peer-reviewing processes of scientific journals and conferences. Naïve Bayes classifier is used – along with word-based tokens – to identify authors and the best result is around 90%. This work is different from ours in several aspects. First, it explores the author identification in a very restricted domain; i.e., academic paper reviews. Second, the number of candidate authors is around 20 which is less than ours(~ 2000). Third, the number of features used in [15] is large where unigram, bigram, and trigrams based on words(a sequence of one, two and three words) are used. In ours, we only use unigrams and bigrams that are based on letters (in addition to the ratings and categories). The work in [3] also considered author identification and similarity detection by incorporating a rich set of stylistic features along with a novel technique(based on Karhunen-Loeve-transforms) to extract write-prints. An identification performance of 91% is achieved. The same approach is tested on a large set of Buyer/Seller Ebay feedback comments collected from Ebay. Such comments typically reflect one's experience when dealing with a buyer or a seller. Unlike our general-purpose reviews, these comments do not review products, services or places of different categories. Additionally, the scale of the problem is different and the analysis is performed for only 100 authors. An author identification technique based on frequent pattern write prints is shown in [9] and author identification techniques based on extracting lexical, syntactic, structural and content-specific features and then feeding them to some classifiers are shown in [21]. For a comprehensive overview of authorship analysis studies, we refer to [19].

While many of the author identification studies are somewhat similar to our present work, there are some notable differences. First, we perform authorship identification analysis in a context that has not been extensively explored – generic user reviews. User reviews are generally are less formal and less restricting in the choice of words. In a review, the author generally assesses something and thus the text conveys some evaluation and personal opinions. In addition, reviews contain other non-textual information, such as the ratings and categories of things being reviewed. These types of extra information provide added

[5] Note in [16], the identification accuracy is increased to 80% by not making a guess when there is not enough confidence; however, this does not increase the linkability ratio (recall is low).

leverage(shown in 4.3). Second, our problem formulation is different. We study linkability of reviews in the presence of a large number of prolific contributors where the number of anonymous reviews could be more than one (up to 60 reviews). Whereas, most prior work attempts to identify authors from a small set of authors, each with small sets of texts. Third, we show high linkability ratios in the presence of very simple features. For example, reviewers can be accurately identified from their letter distributions. These measurement results are very alarming for users concerned about their privacy.

Some work is done in recovering authors based on their ratings, using external knowledge such as [7] and [17]. Another related research effort assesses authenticity of reviews [10]. A related study in [4] proposes linguistic-based techniques to detect user attempts to hide their writing styles.

7 Conclusion

Large numbers of Internet users are becoming frequent visitors and contributors to various review sites. At the same time, they are concerned about their privacy. In this paper, we study linkability of reviews. Based on a large set of reviews, we show that a high percentage (99% in some cases) are linkable, even though we use very simple models and very simple features set. Our study suggests that users reliably expose their identities in reviews, which could be partly due to the way the users write their reviews and the places they select to review. This has certain important implications for cross-referencing accounts among different review sites and detecting people who write reviews under different identities. Additionally, techniques used in this study could be adopted by review sites to give contributors feedback about linkability of their reviews.

References

[1] Yelp By The Numbers. `http://officialblog.yelp.com/2010/12/2010-yelp-by-the-numbers.html`.

[2] Yelp Elite Squad. `http://www.yelp.com/faq#what_is_elite_squad`.

[3] A. Abbasi and H. Chen. Writeprints: A Stylometric Approach to Identity-Level Identification and Similarity Detection in Cyberspace. In *ACM Transactions on Information Systems*, 2008.

[4] S. Afroz, M. Brennan, and R. Greenstadt. Detecting Hoaxes, Frauds, and Deception in Writing Style Online. In *IEEE Symposium on Security and Privacy*, 2012.

[5] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[6] K. Dave, S. Lawrence, and D. M. Pennock. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. In *international conference on World Wide Web*, 2003.

[7] D. Frankowski, D. Cosley, S. Sen, L. Terveen, and J. Riedl. You Are What You Say: Privacy Risks of Public Mentions. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2006.

[8] M. Hu and B. Liu. Mining and Summarizing Customer Reviews. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

[9] F. Iqbal, H. Binsalleeh, B. Fung, and M. Debbabi. A unified data mining solution for authorship analysis in anonymous textual communications. In *Information Sciences (INS): Special Issue on Data Mining for Information Security*, 2011.

[10] N. Jindal and B. Liu. Opinion Spam and Analysis. In *ACM International Conference on Web Search and Data Mining*, 2008.

[11] N. Jindal, B. Liu, and E.-P. Lim. Finding Unusual Review Patterns Using Unexpected Rules. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 2010.

[12] D. Lewis. Naive(bayes) at forty:the independence assumption in information retrieval. In *Proceedings of the 10th European Conference on Machine Learning*, 1998.

[13] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. Lauw. Detecting Product Review Spammers using Rating Behaviors. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 2010.

[14] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[15] M. Nanavati, N. Taylor, W. Aiello, and A. Warfield. Herbert West – Deanonymizer. In *6th USENIX Workshop on Hot Topics in Security*, 2011.

[16] A. Narayanan, H. Paskov, N. Z. Gong, J. Bethencourt, E. Stefanov, E. C. R. Shin, and D. Song. On the Feasibility of Internet-Scale Author Identification. In *IEEE Symposium on Security and Privacy*, 2012.

[17] A. Narayanan and V. Shmatikov. Robust De-anonymization of Large Sparse Datasets. In *IEEE Symposium on Security and Privacy*, 2009.

[18] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment Classification using Machine Learning Techniques. In *Empirical Methods on Natural Language Processing Conference*, 2002.

[19] E. Stamatatos. A Survey of Modern Authorship Attribution Methods. In *Journal of the American Society for Information Science and Technology*, 2009.

[20] S. Yadav, A. K. Reddy, A. N. Reddy, and S. Ranjan. Detecting Algorithmically Generated Malicious Domain Names. In *Internet Measurement Conference*, 2010.

[21] R. Zheng, J. Li, H. Chen, and Z. Huang. A Framework for Authorship Identification of Online Messages: Writing Style Features and Classification Techniques. In *Journal of the American Society for Information Science and Technology*, 2006.