How To Break Anonymity of the Netflix Prize Dataset

Arvind Narayanan and Vitaly Shmatikov

The University of Texas at Austin

March 2, 2007

Abstract. As part of the Netflix Prize contest, Netflix—the world's largest online movie rental service—publicly released a dataset containing movie ratings of 500,000 Netflix subscribers. The dataset is intended to be anonymous, and all personally identifying information has been removed. We demonstrate that an attacker who knows only a little bit about an individual subscriber can easily identify this subscriber's record if it is present in the dataset, or, at the very least, identify a small set of records which include the subscriber's record. This knowledge need not be precise, *e.g.*, the dates may only be known to the attacker with a 14-day error, the ratings may be known only approximately, and some of the ratings may even be completely wrong.
Using the Internet Movie Database (IMDb) as our source of auxiliary information, we successfully identified Netflix records of *non-anonymous* IMDb users, uncovering information—such as their apparent political preferences—that could not be determined from their public IMDb ratings. We also discuss the implications that a successful deanonymization of the Netflix dataset may have for the Netflix Prize competition.

1 Introduction

> *"There's something magical about spying via Netflix. Unlike the fantasy worlds of MySpace and the blogs, it's less a social platform than a practical tool, so the data is exceptionally pure. Netflix allows our tastes to flourish in their full, omnivorous, complex human glory, free of shameful image-management and the high/low divide. Earnest Goes to Camp consorts freely with Citizen Kane. It's like a self-portrait in movie titles: Nowhere else is cultural desire so nakedly on display."*
> – *Sam Anderson in* Slate*, September 14, 2006.*

On October 2, 2006, Netflix, the world's largest online DVD rental service, announced the $1-million Netflix Prize for improving their movie recommendation service [4]. To aid contestants, Netflix publicly released a dataset containing $100,480,507$ movie ratings, created by $480,189$ Netflix subscribers between December 1999 and December 2005.

While movie ratings are not as sensitive as, say, medical records, release of massive amounts of data about individual Netflix subscribers raises interesting privacy issues. Among the Frequently Asked Questions on the Netflix Prize webpage [8], there is the following question: "Is there any customer information in the dataset that should be kept private?" Netflix answers this question as follows:

> No, all customer identifying information has been removed; all that remains are ratings and dates. This follows our privacy policy, which you can review here. Even if, for example, you knew all your own ratings and their dates you probably couldn't identify them reliably in the data because only a small sample was included (less than one-tenth of our complete dataset) and that data was subject to perturbation. Of course, since you know all your own ratings that really isn't a privacy problem is it?

In general, removing the identifying information from the data is not sufficient for anonymity. The attacker may be able to join the (ostensibly) anonymized dataset with auxiliary data, resulting in a complete breach of privacy. This has been demonstrated, for example, by Sweeney, who deanonymized medical records by joining them with a publicly available voter database [9], as well as by privacy breaches caused by publicly released AOL search data [5]. In the case of the Netflix movie ratings dataset, the attacker may already know a little bit about some subscriber's movie preferences: the titles of a few of the movies that this subscriber watched, whether she liked them or not, maybe even approximate dates when she watched them.

Anonymity of the Netflix dataset thus depends on the answer to the following question: **How much does the attacker need to know about a Netflix subscriber in order to identify her record in the dataset, and thus learn her complete movie viewing history?**

In the rest of this paper, we investigate this question. In brief, the answer is: very little. For example, suppose the attacker learns a few random ratings and the corresponding dates for some subscriber. We expect that the dates when the ratings are entered into the Netflix system are strongly correlated with the dates when the subscriber actually watched the movies, and can thus be inferred by the attacker. To account for the imprecision of date knowledge, in our analysis the attacker's knowledge of the dates has either a 3-day, or 14-day error.

With 8 movie ratings (of which 2 may be completely wrong) and dates that may have a 3-day error, 96% of Netflix subscribers whose records have been released can be uniquely identified in the dataset. For 64%, *two* ratings and dates are sufficient for complete deanonymization, and for 89%, two ratings and dates are enough to reduce the set of candidate records to 8 out of almost 500,000, which can then be inspected by a human for further deanonymization. If the movies in question are *not* among the top 100 most rated, then even with a 14-day error in the dates, approximate knowledge of 8 ratings (2 of which are wrong) completely de-anonymizes 80% of the subscribers in the dataset.

Does privacy of Netflix ratings matter? One may wonder whether learning someone's entire movie viewing history should be considered a serious privacy breach. We admit that we do not know what percentage of Netflix subscribers view their movie ratings as sensitive, since conducting such a survey does not appear feasible without Netflix's cooperation.

We emphasize that the privacy question is *not* "Does the average Netflix subscriber care about the privacy of his movie viewing history?" The right question is "Are there *any* Netflix subscribers whose privacy can be compromised by analyzing the Netflix Prize dataset?" The answer to the latter question is, undoubtedly, yes. As our experiments with cross-correlating public IMDb records with anonymized Netflix records show, it is possible to learn sensitive *non-public* information about a person's political or even sexual preferences. We assert that even if the vast majority of Netflix subscribers didn't care about the privacy of their movie ratings (which is not obvious by any means), the analysis presented in this paper would still indicate serious privacy issues with the Netflix Prize dataset.

Moreover, the linkage between an individual and her movie viewing history has implications for her *future* privacy. In network security, "forward secrecy" is important: even if the attacker manages to compromise a session key, this should not help him much in compromising the keys of future sessions. Similarly, one may state the "forward privacy" property: if someone's privacy is breached (*e.g.*, her anonymous online records have been linked to her real identity), future privacy breaches should not become easier. Now consider a Netflix subscriber (call her Alice) whose entire movie viewing history has been revealed using one of the techniques we present in the paper. If in the future Alice creates a brand-new virtual identity (call her Ecila) then, even under the cover of perfect anonymity, Ecila can *never* disclose any non-trivial information about her movie viewing, because any such information can be traced back to her real identity via the Netflix Prize dataset. In general, once any piece of data has been

linked to a person's *real* identity, any association between this data and an anonymous *virtual* identity breaks anonymity of the latter.

It also appears that Netflix might be in violation of its own stated privacy policy. According to this policy, "Personal information means information that can be used to identify and contact you, specifically your name, postal delivery address, e-mail address, payment method (e.g., credit card or debit card) and telephone number, as well as other information when such information is combined with your personal information. [...] We also provide analyses of our users in the aggregate to prospective partners, advertisers and other third parties. We may also disclose and otherwise use, on an anonymous basis, movie ratings, commentary, reviews and other non-personal information about customers." The simple-minded division of information into personal and non-personal is a false dichotomy, as we demonstrate in the rest of this paper.

Obtaining auxiliary information. Given how little auxiliary information is needed to deanonymize subscriber records in the Netflix dataset, a determined attacker should not find it difficult to obtain such information, especially since it need not be precise. A water-cooler conversation with an office colleague about her cinematographic likes and dislikes may yield enough information, especially if at least a few of the movies mentioned are outside the top 100 most rated Netflix movies. This information can also be gleaned from personal blogs, Google searches, and so on.

An excellent source of users' movie ratings is the Internet Movie Database (IMDb) [7]. We expect that there is a very strong correlation between a Netflix subscriber's Netflix ratings, and his IMDb ratings. We believe that for many Netflix subscribers who are active on the IMDb, it may be possible to identify their records in the Netflix database (provided they are among the subscribers whose data have been released). We demonstrate feasibility of this approach in section 4.

What Netflix did or should have done with the data. According to the Netflix Prize FAQ [8], the ratings data have been perturbed. By having some of our friends share their ratings history from the Netflix website with us, we found that some records have been removed, but no perturbation has been applied to the remaining records (see appendix A).

It is also interesting to consider the Netflix Prize dataset from the perspective of k-anonymity [10]. A dataset is k-anonymous if every record is "indistinguishable" from at least $k-1$ other records. k-anonymity fundamentally relies on locality of data, and all known techniques fail on high-dimensional data [1]. In fact, the expected distances to the nearest

and farthest neighbors are almost the same for high-dimensional data under a variety of distributions [2]. Our analysis of the Netflix dataset confirms this phenomenon: the records are extremely *sparse*, *i.e.*, with an overwhelming probability a random subscriber's record has no neighbors whose records are similar. Consider each record as a vector in a multi-dimensional space, where each movie corresponds to a dimension, and the subscriber's coordinates are his or her ratings on the movies. Even if we ignore numerical ratings and simply consider binary coordinates (1 if the subscriber rated a given movie, 0 if he didn't), the vast majority of subscribers have *no* neighbors within 70% Hamming distance of their record (figure 1). This implies that (i) in all likelihood, the dataset has *not* been k-anonymized, and (ii) even 2-anonymization would destroy most of the information contained in the dataset. These observations are in line with those of [3] (see below.)

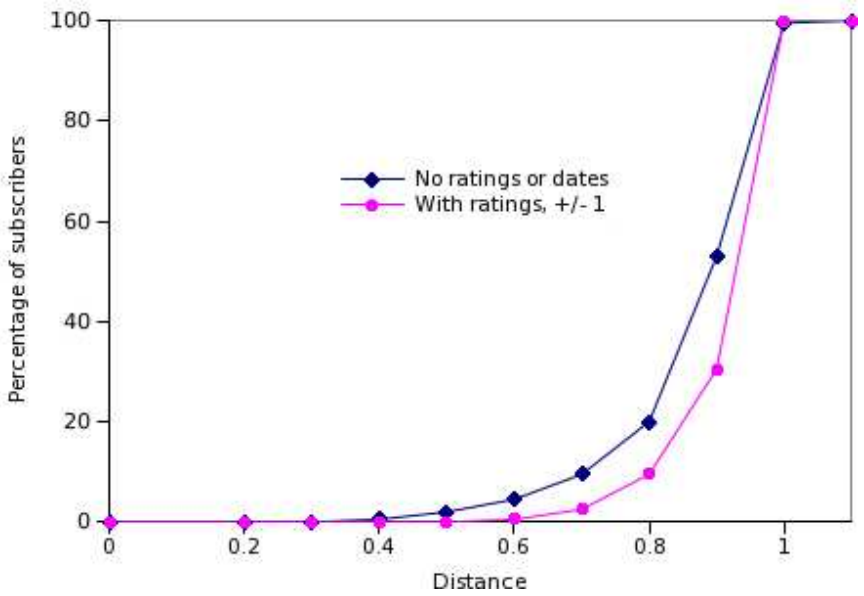

Fig. 1. Distance to nearest neighbor (as fraction of Hamming weight). Note that the distance can be more than 1.

Matt Wright suggested in a personal communication that Netflix should have withheld movie names. While this would make deanonymization much harder (although not obviously impossible), it is not clear to what extent it would affect utility of the data. While Cinematch, Netflix's

current recommendation algorithm, does not make use of movie metadata, we do not know what the competing algorithms do.

Related work. We are aware of only one other research paper that considers privacy of movie ratings [3] (research described in this paper was done independently, without knowledge of [3]). Working in collaboration with the MovieLens movie recommendation service, Frankowski *et al.* demonstrated in [3] that it is possible to correlate public mentions of movies in the MovieLens discussion forum with the MovieLens users' movie rating histories in the *internal* MovieLens dataset. By contrast, this paper is based only on the publicly available data. Netflix did not cooperate with us in any way. Unlike the MovieLens analysis, our de-anonymization is *not* based on cross-correlating Netflix internal datasets (to which we do not have access) with public Netflix forums. All of our analyses can be performed by a complete outsider.

Moreover, our analyses require much less auxiliary information about a user than [3]. While Frankowski *et al.* investigate what fraction of users can be de-anonymized given *all* of their public ratings (29 per user, on average), we show that de-anonymization is possible with knowledge of only 2 to 8 ratings. Finally, our de-anonymization algorithm is *robust* in the presence of false (some of the ratings "known" to the attacker can be completely wrong) and fuzzy auxiliary information. By contrast, it appears that the algorithm of [3] can be foiled simply by publicly mentioning a few movies that one hasn't rated in the private dataset.

It is also worth observing that the Netflix Prize dataset has privacy implications for a much larger set of people. It contains the movie viewing histories of 500,000 subscribers; by contrast, the largest publicly available MovieLens dataset contains the data of only 6,000 users.

Organization of the paper. In section 2, we give formal definitions of dataset anonymity. In section 3, we analyze anonymity of the Netflix Prize dataset. In section 4, we demonstrate how our techniques work, using the Internet Movie Database as the source of auxiliary information. In section 5, we discuss future research and the implications of our results for the Netflix Prize competition.

2 Anonymity: formal definitions

Let $\mathcal{M}$ be the set of all movies in the Netflix Prize dataset, and $\mathcal{C}$ be the set of subscribers who records have been released. Suppose the attacker is interested in discovering the record of a particular person $c \in \mathcal{C}$ in the dataset, and that he knows some subset of movies $M \subset \mathcal{M}$ that his

victim has watched and rated. We will refer to this subset as the attacker's *auxiliary information*.

For each movie $m \in M$, let $r_c(m)$ and $d_c(m)$ be, respectively, the victim's rating (on the 1-to-5 scale) and the date it was entered into the Netflix system. Suppose that, for each movie m, the attacker knows the rating $\hat{r}_c(m)$ and the rating date $\hat{d}_c(m)$. We emphasize that the attacker's knowledge need not be precise, *i.e.*, we do not require that $r_c(m) = \hat{r}_c(m)$.

Formally, precision of the attacker's knowledge is controlled by parameters $\epsilon_r, \epsilon_d, \delta_r$, and δ_d such that

$$\mathbf{P}_{m \in M}[|r_c(m) - \hat{r}_c(m)| \leq \epsilon_r] \geq 1 - \delta_r$$
$$\mathbf{P}_{m \in M}[|d_c(m) - \hat{d}_c(m)| \leq \epsilon_d] \geq 1 - \delta_d$$

In other words, for some subset M of the movies watched by the victim, the attacker knows the victim's ratings (respectively, dates) within some error ϵ_r (resp., ϵ_d). On the δ_r (resp., δ_d) fraction of the movies known to the attacker, the attacker's knowledge is permitted to be completely incorrect. For example, if the attacker has no information about dates whatsoever, then $\delta_d = 1$.

The set of movies M watched by the victim c and known to the attacker is sampled from some distribution over $\mathcal{M}$, defined by the predicate $\pi : \mathcal{M} \to \{0, 1\}$ and the size parameter k. M is a random variable which consists of k samples drawn uniformly without replacement from the set $\{m \in \mathcal{M} : \pi(m) = 1 \text{ and } r_c(m) \neq \perp\}$. For example, if $\forall m \in \mathcal{M}\ \pi(m) = 1$, then we put no restrictions on the movies known to the attacker.

We will also consider scenarios in which the attacker knows the victim's ratings on some movies that are not among the top 100 and top 500 most frequently rated Netflix movies. Because each such movie is rated by relatively few people, deanonymization requires less auxiliary data.

Definition of anonymity breach. The attacker's goal is to identify the record of his victim among the 480,189 subscriber records in the Netflix dataset. This identification need not be precise in order to be considered a serious privacy breach. For example, the attacker may be able to identify two subscriber records and determine with certainty that one of them is the victim's. Once the set of candidate records is small, deanonymization can be completed with manual analysis of the ratings and additional auxiliary information, *e.g.*, knowledge of when the subscriber joined Netflix.

Define the *neighborhood* $N(c)$ of a subscriber c as

$$N_M(c) := \{c' : \mathbf{P}_{m \in M}[|r_{c'}(m) - \hat{r}_c(m)| \leq \epsilon_r] \geq 1 - \delta_r \text{ and } \mathbf{P}_{m \in M}[|d_{c'}(m) - \hat{d}_c(m)| \leq \epsilon_d] \geq 1 - \delta_d\}$$

Informally, the neighborhood is the set of subscribers $\{c'\}$ who "look like" the targeted subscriber c. Define $n_M(c) = |N_M(c)|$. For example, if $n_M(c) = 1$, this means the attacker can completely identify c's record in the database on the basis of M movies whose ratings by c he (approximately) knows. This corresponds to a complete anonymity breach.

If $n_M(c) = 10$, then the attacker can identify a "lineup" of 10 records, one of which must be the victim's record, and so on. We will refer to this set as the set of *candidate records*. The size of this set gives us a metric for how much anonymity of a single subscriber is reduced, given some auxiliary information M. Define

$$N_{\pi,k}(c) = N_{M \leftarrow \{m \in M : \pi(m)=1 \text{ and } r_c(m) \neq \perp\}, |M|=k}(c)$$

and

$$n_{\pi,k}(c) := |N_{\pi,k}(c)|$$

Here $n_{\pi,k}$ is the random variable that measures the size of a random Netflix subscriber's "neighborhood" given a particular π (*i.e.*, condition on the movies whose ratings and dates are approximately known to the attacker) and k (the number of such movies).

Finally, define

$$\mu(n, k) := \mathbf{P}_{c \in \mathcal{C}}[n_{\pi,k}(c) \leq n]$$

This is the probability that a random subscriber has her "neighborhood" reduced to n on the basis of k movie ratings approximately known to the attacker. We will say that this subscriber has been *n-deanonymized* with probability μ. We reiterate that small values of n imply bigger anonymity loss, with $n = 1$ implying complete identification.

3 Anonymity of the Netflix dataset

We analyzed the released Netflix dataset with respect to anonymity definitions given in section 2. Each data point in this section is based on 100 records sampled uniformly and independently from the dataset.

Attacker knows ratings and dates. Suppose the attacker knows the subscriber's ratings on M random movies among those watched by the subscriber, along with associated dates (with a 3-day error, *i.e.*, $\epsilon_d = 3$). Some of the attacker's knowledge may be incorrect. We will say that the attacker knows P *out of* M if only P of the ratings are correct. Formally, this corresponds to $\delta_r = \delta_d = \frac{M-P}{M}$ and $\epsilon_r = 0$.

We will look at the scenarios where the attacker knows 2 ratings, 3 out of 4, and 6 out of 8. Because the average subscriber has 214 ratings, this

represents a negligible information leakage. Results are shown in fig. 2. In this and all subsequent figures, each curve corresponds to a scenario in which the attacker knows M ratings, but only P of them correctly, $\delta_r = \delta_d = \frac{M-P}{M}$, and $\mu(n, M)$ is shown on the Y axis.

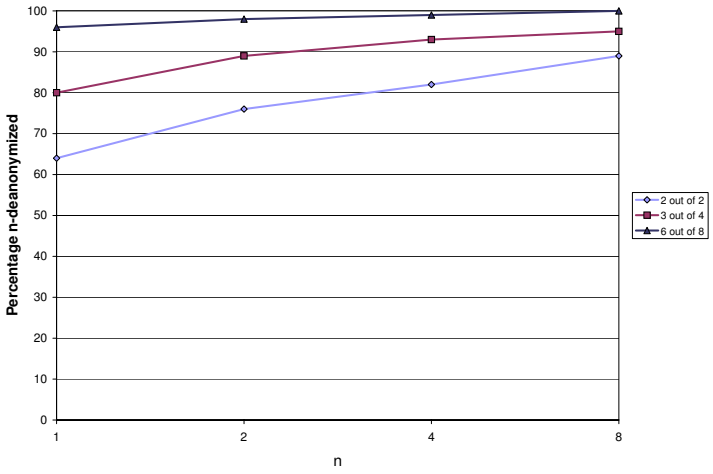

Fig. 2. Attacker knows precise ratings and dates with a 3-day error for 2, 4, or 8 movies in the subscriber's history, some of them incorrectly.

Our analysis shows that there is very little anonymity in the Netflix dataset. For example, knowledge of ratings and dates on only 2 random movies from the subscriber's history completely deanonymizes 64% of subscribers in the dataset. Another 25% cannot be identified precisely, but the attacker can reduce the set of candidate records, one of which is the subscriber's true record, to as few as 8 records. In combination with manual analysis, this may very well lead to complete deanonymization. Knowledge of ratings on 8 random movies, even if 2 of these ratings are completely wrong, deanonymizes 96% of the subscribers in the dataset.

Attacker approximately knows ratings and dates on less popular movies. Now suppose the attacker knows the subscriber's ratings on movies other than the top 100 most rated movies. We deliberately make the attacker's knowledge very fuzzy. If the true rating is r_c, the rating known to the attacker is selected uniformly from $\{r_c-1, r_c, r_c+1\}$. (recall that ratings are on the 1-5 scale). Formally, we set $\epsilon_r = 1$. We also allow a 14-day error in the date, *i.e.*, $\epsilon_d = 14$. The rating and date errors are independent, *i.e.*, some ratings may have the rating error, and others may

have the date error. Even this very imprecise knowledge is sufficient to deanonymize a large fraction of records, as shown in fig. 3.

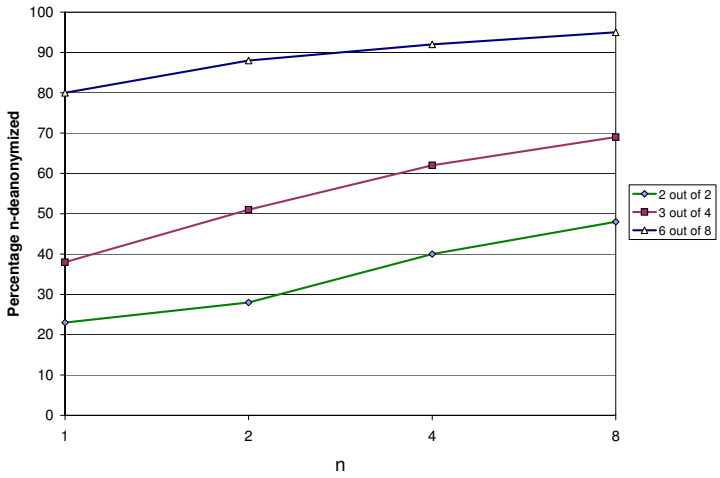

Fig. 3. Attacker knows approximate ratings and dates with a 14-day error for 2, 4, or 8 movies in the subscriber's history, some of them incorrectly. The movies are not among the top 100 most rated movies.

Knowledge of 6 out of 8 ratings (this means the attacker knows 6 ratings within the ± 1 error, and also incorrectly thinks he "knows" 2 additional ratings) deanonymizes 80% of subscribers. Even with as few as 2 imprecise ratings, for almost half of the subscribers the attacker can reduce the set of candidate records to 8 out of $500,000$.

The assumption that a given subscriber has rated movies that are not among the top 100 most rated movies may appear implausible, because "most people watch only blockbusters." This does not appear to be the case. For example, 93% of subscribers in the released dataset rated at least 10 movies outside the top 100 most rated movies. Therefore, with overwhelming probability, the attacker's chosen victim has rated a significant number of movies that are not in the top 100 most rated movies.

	Percentage of subscribers who rated ...		
	At least 1 movie	At least 5	At least 10
Not in 100 most rated	100%	97%	93%
Not in 500 most rated	99%	90%	80%
Not in 1000 most rated	97%	83%	70%

Attacker does not know the dates. Even approximate knowledge of dates is very useful, and removing the dates from the attacker's knowledge reduces success of deanonymization. Results for this case ($\delta_d = 1$) are shown in fig. 4. Even without the dates, knowledge of the ratings alone on 6 movies that are not in the top 100 most rated movies reduces the plausible record set to 8 records for 69% of subscribers. Knowing 6 correct and 2 incorrect ratings on movies that are not in the top 500 most rated movies completely deanonymizes 46% of subscribers.

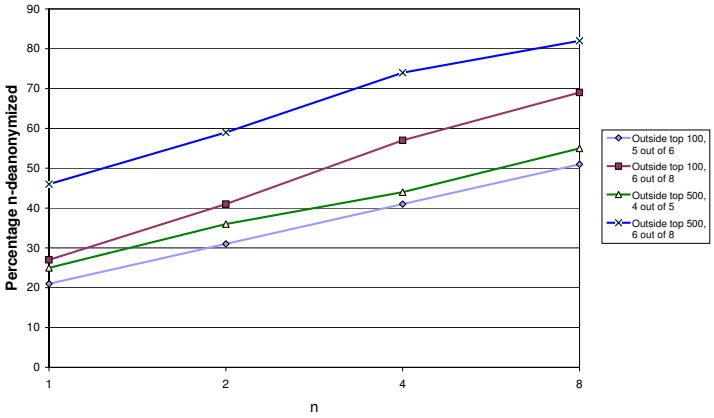

Fig. 4. Attacker knows the ratings, but not the dates, for 5, 6, or 8 movies in the subscriber's history, some of them incorrectly. The movies are not among the top 100 (resp., 500) most rated movies.

4 De-anonymization based on IMDb information

We now describe how to use publicly available auxiliary information from the Internet Movie Database (IMDb) to find the records of IMDb users in the Netflix Prize dataset. Due to the restrictions on crawling IMDb imposed by IMDb's terms of service (note that a real attacker may not comply with these restrictions), we worked with a very small sample of a few dozen IMDb users. Results presented in this section should be viewed as a proof of concept. They do not imply anything about the percentage of IMDb users who can be identified in the Netflix Prize dataset.

The attacks are slightly different in character from what we have described so far. For example, the data are noisier in several ways:

– A large fraction of the movies rated on IMDb are not in Netflix, and vice versa, *e.g.*, movies that have not been released in the US.
– Some of the ratings on IMDb are missing (*i.e.*, the user entered only a comment, not a numerical rating). Note that such data are still useful for deanonymization because not all users watch all movies.
– IMDb users among Netflix subscribers fall into a continuum of categories with respect to rating dates, separated by two extremes: some meticulously rate movies on both IMDb and Netflix at the same time, and others rate them whenever they have free time (which means the dates may not be correlated at all).

Somewhat offsetting these disadvantages is the fact that we can use all of the user's ratings publicly available on IMDb.

To prove that deanonymization works, we need an algorithm with an extremely low false positive rate, because we have no "oracle" to tell us whether the record our algorithm has found in the Netflix Prize dataset based on the ratings of some IMDb user indeed belongs to that user.

Suppose there is a function Δ which measures the correlation between the IMDb record and the Netflix record. Intuitively, Δ should be thought of as the reciprocal of a distance function. Let $|c|$ denote the number of movies rated by the subscriber c on Netflix. Given an IMDb user c_{IMDb}, we try to find a Netflix subscriber $c_{Netflix}$ such that, for some $\lambda > 0$,

$$\forall c' \in C_{Netflix} \text{ s.t. } |c'| < |c| :$$
$$\Delta(c_{Netflix}, c_{IMDb}) \geq \Delta(c'_{Netflix}, c_{IMDb}) + \lambda \sigma_{c'' \in C_{Netflix}, |c''| \leq |c|}(\Delta(c''_{Netflix}, c_{IMDb}))$$

The intuition is that the correlation between the best match and the second-best match for the IMDb user in the Netflix dataset is more than λ times the standard deviation of the correlation. The higher the λ, the higher the confidence that we have indeed found the Netflix record corresponding to c_{IMDb}. The reason we only compare each subscriber with other subscribers who have at most that many ratings is, intuitively, that Netflix subscribers with a large number of ratings are dramatically more likely to be false positives.

The class of functions we considered as candidates for Δ was:

$$\Delta(c, c') = \sum_m \alpha + \beta \cdot 2^{-\frac{|r_m(c) - r_m(c')|}{r_0}} + \gamma \cdot 2^{-\frac{|d_m(c) - d_m(c')|}{d_0}}$$

where the sum is taken over movies m rated by both c and c'. We empirically chose $\alpha = 0, \beta = 1, \gamma = 2, r_0 = 1, d_0 = 14$ for the constants, since these values produced the best results in our experiments.

The above algorithm identified two users from our small IMDb sample in the Netflix dataset with λ of around 28 and 15, respectively (*i.e.*, the Netflix records in question are 28 and 15 standard deviations away from the second-best candidates). Interestingly, the first user was deanonymized mainly from the ratings and the second mainly from the dates.

At this point, although we can assert definitively that deanonymization is possible, it remains more of an art than a science. For example, we are currently predicting the missing ratings to be the mean of the user's all other ratings, but there may be better alternatives.

Let us summarize what our algorithm achieves. Given a user's *public* IMDb ratings, which the user posted voluntarily to selectively reveal *some* of his (or her; but we'll use the male pronoun without loss of generality) movie likes and dislikes, we discover *all* the ratings that he entered *privately* into the Netflix system, presumably expecting that they will remain private. A natural question to ask is why would someone who rates movies on IMDb—often under his or her real name—care about privacy of his movie ratings? Consider the information that we have been able to deduce by locating one of these users' entire movie viewing history in the Netflix dataset and that *cannot* be deduced from his IMDb ratings.

First, we can immediately find his political orientation based on his strong opinions about "Power and Terror: Noam Chomsky in Our Times" and "Fahrenheit 9/11." Strong guesses about his religious views can be made based on his ratings on "Jesus of Nazareth" and "The Gospel of John". He did not like "Super Size Me" at all; perhaps this implies something about his physical size? Both items that we found with predominantly gay themes, "Bent" and "Queer as folk" were rated one star out of five. He is a cultish follower of "Mystery Science Theater 3000". This is far from all we found about this one person, but having made our point, we will spare the reader further lurid details.

5 Implications for the Netflix Prize and future work

Deanonymization of Netflix subscribers may enable one to learn the true ratings for some entries in the Netflix Prize *test* dataset (these ratings have been kept secret by Netflix). The test dataset has been chosen in such a way that the contribution of any given subscriber is no more than 9 entries (see fig. 5). Therefore, it is not possible to find a small fraction of subscribers whose ratings will reveal a large fraction of the test dataset.

Access to true ratings on the test dataset does not translate to an immediate strategy for claiming the Netflix Prize. The rules require that

the algorithm be submitted for perusal. In spite of this, having the test data (or the data closely correlated with the test data) enables the contestant to train on the test data in order to "overfit" the model. This is why Netflix kept the ratings on the test data secret.

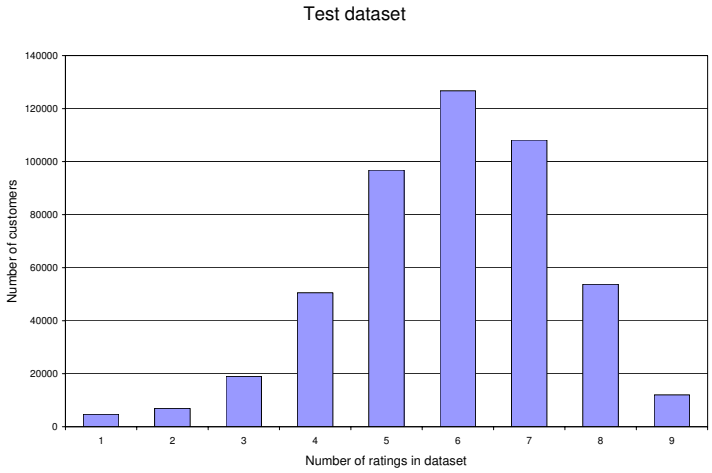

Fig. 5. Test dataset for the Netflix Prize.

How many Netflix subscribers would need to be deanonymized before there is a significant impact on the performance of a recommendation algorithm that uses this information? The root mean squared errors (RMSE) of the current top performers (as of March 2, 2007) are about 0.89. If a subscriber's "true" ratings are available, the error for that subscriber drops to zero. Thus, if the learner has access to $\frac{1}{0.89} = 1.12\%$ of deanonymized records, then the RMSE score improves by 1% (assuming that the contribution of each subscriber is the same and RMSE behaves roughly linearly). This is roughly equal to the difference the current 1st and 20th contestants on the Netflix Prize leader board.

How easy is it to deanonymize 1% of the subscribers? The potential sources of large-scale true rating data are the publicly available ratings on the site itself, IMDb, and the subscribers themselves. Netflix appears to have taken the elementary precaution of removing from the dataset the ratings of the subscribers that are publicly available on the Netflix website. While our experiments in section 4 show that successful cross-correlation of IMDb and Netflix records is possible, there are some hurdles to overcome: it is not clear what fraction of users with a significant body

of movie ratings on IMDb are also Netflix subscribers, nor is it known how ratings and dates on IMDb correlate with those on Netflix for the average user (although we expect a strong correlation).

Collecting data from the subscribers themselves appears to be the most promising direction. Many Netflix subscribers do not regard their ratings as private data and are eager to share them, to the extent that there even exists a browser plugin that automates this process, although we have not found any public rating lists generated this way. If the here-are-all-my-Netflix-ratings "meme" propagates through the "blogosphere," it could easily result in a publicly available dataset of sufficiently large size. It is also easy for a malicious person to bribe subscribers (say, "upload your Netflix ratings to gain access to the protected areas of this site"). Also, many subscribers have "friends" on Netflix, and subscribers' ratings are accessible to their friends.

We emphasize that even though many Netflix subscribers do not regard their movie viewing histories as sensitive, this does *not* mean that privacy of Netflix records is moot. In section 4, we extracted from the Netflix Prize dataset non-public information about some subscribers that should be considered sensitive by any reasonable definition.

A possible direction for future work is to extract social relationships between Netflix users based on their movie rating histories. Consider two friends, both Netflix subscribers, who tend to watch movies together. The dates on some subsets of their respective ratings will be strongly correlated (even though their ratings might not be). Of course, one would see spurious correlations even for unrelated subscribers, because movies tend to be watched more right after they are released, and so on. Nevertheless, one can correct for such effects by looking at the entire body of subscribers. Knowledge of social clusters obtained in this manner can be a source of information for further deanonymization [6], but detailed explanation is beyond the scope of this paper.

References

1. Charu Aggarwal. On k-anonymity and the curse of dimensionality. In *VLDB*, 2005.
2. Kevin Beyer, Jonathan Goldstein, Raghu Ramakrishnan, and Uri Shaft. When is "nearest neighbor" meaningful? *Lecture Notes in Computer Science*, 1540:217–235, 1999.
3. D. Frankowski, D. Cosley, S. Sen, L. Terveen, and J. Riedl. You are what you say: privacy risks of public mentions. In *Proc. 29th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 565–572. ACM, 2006.

4. K. Hafner. And if you liked the movie, a Netflix contest may reward you handsomely. New York Times, Oct 2 2006.
5. S. Hansell. AOL removes search data on vast group of web users. New York Times, Aug 8 2006.
6. B. Hayes. Connecting the dots: Can the tools of graph theory and social-network studies unravel the next big plot? *American Scientist*, September–October 2006. `http://www.americanscientist.org/template/AssetDetail/assetid/53062`.
7. IMDb. The Internet Movie Database. `http://www.imdb.com/`, 2007.
8. Netflix. Netflix Prize: FAQ. `http://www.netflixprize.com/faq`, Downloaded on Oct 17 2006.
9. L. Sweeney. Weaving technology and policy together to maintain confidentiality. *J. of Law, Medicine and Ethics*, 25(2–3):98–110, 1997.
10. L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty*, 10(5):571–588, 2002.

A On perturbation of the Netflix dataset

Figs. 6 and 7 plot the number of ratings X against the number of subscribers in the released dataset who have at least X ratings. The tail is surprisingly thick: thousands of subscribers have rated more than a thousand movies. Netflix claims that the subscribers in the released dataset have been "randomly chosen." Whatever the selection algorithm was, it was not uniformly random. Common sense suggests that with uniform subscriber selection, the curve would be monotonically decreasing (as most people rate very few movies or none at all), and that there would be no sharp discontinuities.

It is not clear how the data was sampled. Our conjecture is that some fraction of the subscribers with more than 20 ratings were sampled, and the points on the graph to the left of $X = 20$ are the result of some movies being deleted after the subscribers were sampled.

We requested the rating history as presented on the website from some of our friends, and based on this data, located two of them in the database. Netflix's claim that the data were perturbed does not appear to be borne out. One of the subscribers had 1 of 306 ratings altered, and the other had 5 of 229 altered. (These are upper bounds, because they include the possibility that the subscribers changed the ratings after the snapshot that was released was taken.) In any case, the level of noise is far too small to affect the deanonymization algorithms. We have no way of determining how many dates were altered and how many ratings were deleted, but we suspect that very little perturbation has been applied.

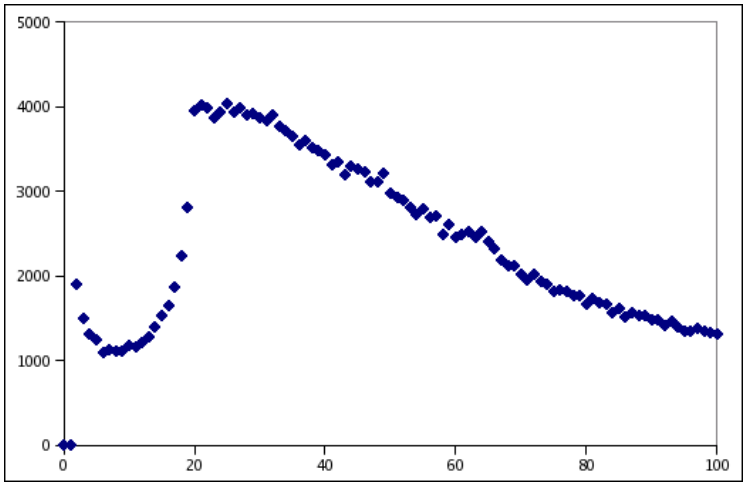

Fig. 6. For each $k \leq 100$, the number of subscribers with k ratings in the released dataset.

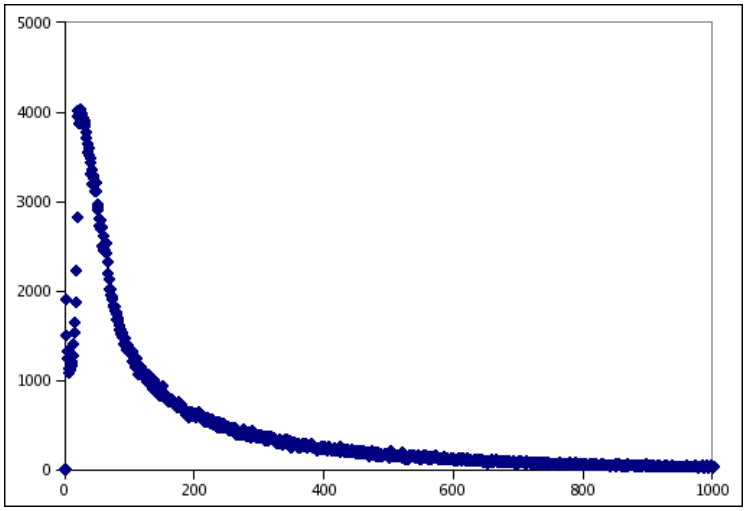

Fig. 7. For each $k \leq 1000$, the number of subscribers with k ratings in the released dataset.