Social Networks and Social Contagion ...
AAAI Technical Report FS-13-05

Associative Patterns of Web Browsing Behavior

Myriam Abramson
Naval Research Laboratory, Code 5584
Washington, DC 20375
myriam.abramson@nrl.navy.mil

Shantanu Gore
Science & Engineering Apprenticeship Program
Naval Research Laboratory, Code 5584
Washington, DC 20375
gore.shantanu@gmail.com

Abstract

As more people use the Web through a browser to gather and disseminate information, recognizing Web browsing signatures can complement other behavioral biometrics such as keystroke authentication to verify a claim of identity and/or identify persons of interest. The deluge of available digital traces enables the cognitive analysis of behavioral traits that differentiate between users and predict their online behavior. Recommendation systems have long capitalized on this capability to personalize search queries but have not exploited the temporal structure of preferences. This paper claims that spatio-temporal patterns of category of website visited by time of access can uniquely characterize and identify users. We present some exploratory approaches in user identification based on recurrent neural networks and empirical results based on clickstream data obtained through a user study and through an internet data provider.

1 Introduction

The problem of user identity is one of the fundamental and still largely unresolved problems of cyberspace, testing the boundary between trust and privacy. Multiple approaches have been proposed to solve this problem through consolidated password schemes (e.g., OpenID (Thibeau and Reed 2009), Firefox's Persona (Mills 2011)). On the other hand, the popularity of social media such as Facebook and Twitter have made possible the availability of large amount of spontaneous online usage behavior ripe for analysis and individual search history patterns are already used by search engines to personalize search results. Reality mining (Pentland and Pentland 2008) captures unconscious patterns of behavior through signals obtained from wearable mobile computing devices to reveal personal characteristics in order to shape human interaction. As our interaction with the Web becomes more natural and even mediates our interaction with others (Turkle 2012), we claim that Web browsing behavior can be rich enough to uniquely characterize who we are through unconscious behavioral patterns and authenticate ourselves with a cognitive fingerprint.

Attribution is broadly defined as the assignment of an effect to a cause. We differentiate between authentication and identification as two techniques for the attribution of identity. Authentication is defined as the verification of claimed identification (Jain, Bolle, and Pankanti 1999). Identification involves recognition as a one-to-many matching problem while authentication is a one-to-one matching problem. It is possible to do identification with a series of authentication procedures. Likewise, authentication can be obtained through identification with a reject option. This paper focuses on identification.

The paper is organized as follows. In Section 2, we briefly describe prior research on the modeling of Web browsing behavior and attribution in cyberspace. In Section 3 we present our technical approach starting in Section 3.1 with our descriptive analysis of the different features of Web browsing behavior from clickstream data obtained through a user study. In Section 3.2, we introduce our approach using recurrent neural networks and our motivation for this approach. In Section 4, we present our empirical results from our technical approach on identifying users on two different datasets. Our conclusions and future work suggestions are found in Section 5.

2 Related Work

Marketers have long been interested in understanding Web interaction behavior (Atterer, Wnuk, and Schmidt 2006) in order to design Web sites that entice visitors to finish their Web session with a checkout of their shopping cart. *Behavioral targeting* is an approach used by advertisers (e.g., DoubleClick) that tracks Web behavior to deliver advertisements which match an individual's semantic profile defined by content-related preferences and interests. Research in this area has concentrated on identifying the demographic characteristics of a behavior such as age and gender rather than authenticating a single individual (De Bock and Van den Poel 2010). There has also been some research on understanding online browsing behavior from an aggregate perspective in order to identify influential websites in user navigation patterns (Kumar and Tomkins 2010).

In contrast to semantic patterns, syntactic patterns characterize Web browsing based strictly on session and navigation features. They include the burstiness of pageviews (giving rise to the Slashdot effect), the number of page revisits, and the number of pages between revisits (Kumar and Tomkins 2010). In addition, the length of a session (both time and

number of pages visited), the starting time and day of the week also characterize user syntactic patterns.

The attribution problem in cyberspace has been addressed in several ways mainly by leveraging from features in the browser (e.g., history stealing, cookies) or accessing datasets containing partially identifying information. For example, de-anonymization in social networking websites has been accomplished by computing the intersection of users from group memberships in a social network using information from hyperlinks in the browser history and knowledge about those groups (Wondracek et al. 2010). In general, unique identification is possible by cross-referencing independent information sets containing partial information with a universal set in a manner equivalent to a database join (also known as "linkage attacks"). For example, it has been possible to link medical records to individuals in voter registration records (Sweeney 1996). Some success has been reported with the classification of global syntactic features of a Web session (e.g. length of session, average time on a page) per user (Padmanabhan and Yang 2006) aggregated over several sessions. It has also been shown that authorship of content can be determined from stylometric features on an internet scale threatening anonymity (Narayanan et al. 2012) but this type of attribution depends on published content. Research in predicting user behavior in cyberspace has also been focused on improving tasks such as information retrieval (Armstrong et al. 1995). For example, based on the content of the current webpage and a user's original search keywords, the most relevant hyperlinks in the page are highlighted to guide selection of the next page to visit. This type of prediction is oriented toward the information presented in context to the user rather than the specific activity that a user might pursue (e.g. send an email, read a paper, etc.). In contrast to previous approaches, we address the attribution problem by leveraging both from syntactic patterns in Web browsing history and the semantic content of this history with the genre of the page.

The authentication problem has been addressed in the context of masquerade detection in computer security by modeling user command line sequences. In the masquerade detection problem, the task is to positively identify masqueraders but not to positively identify a particular user. Recent experiments modeling user-issued OS commands as bag-of-words without timing information have obtained a 72.7% true positive rate and a 6.3% false positive rate (Salem and Stolfo 2010) on a set of 15000 commands for 70 users grouped in sets of 100 commands. In that work, a one-class support vector machine (SVM) was shown to produce better performance results than threshold-based comparison with a distance metric. SVMs are a "discriminative" machine learning approach to the identification problem that do not need to have a representation of the user. This approach has been applied to Web browsing behavior (Abramson and Aha 2013) obtaining an average authentication rate of 83% true positive rate and 18% false positive rate for 12 users. In this paper, our proposed approach uses a "generative" machine learning method for identifying a user based on a reconstructed representation of that user.

3 Proposed Approach

We are using Hopfield networks, a recurrent neural net approach, to model the spatio-temporal behavior of a user where the page type visited characterizes a site in a way analogous to a geographical location in mobile behavior. Our motivation for this approach is the capability to represent Web browsing behavior in an organic way across time and space (websites) rather than in a bag-of-words approach. We first describe our user study to obtain clickstream data.

3.1 Web Browsing Modeling

Logging of spontaneous clickstream data in this user study consisted of recording through custom-built browser extensions (Firefox and Chrome) the timestamp and the URL that was visible at the time by the user in the address bar of the browser (i.e., *pageview*). The data was parsed offline to minimize interference with the user. Twelve subjects (3 females and 9 males) participated in this study during the course of their work for one month. For clarity, we only show the results of the same 3 users in our figures. The population was fairly homogeneous and rated themselves highly "Web savvy." The number of pageviews per user varied from 1200 to 12000. Web browsing behavioral data is noisy and requires some pre-processing for analysis. Noise occurs due to distortion from the network behavior, errors in accessing URLs including time-out errors, and automatic page insertion in the browser. Future work will mitigate those problems. We categorized the URL into page types using Diffbot[1], a web service for genre classification. Future work will develop a genre "palette" that enhances the attribution from Web browsing behavior. For example, distinguishing documents between articles and blogs might help distinguish their readers. Other features were extracted from the data: day-of-week, time-of-day, pauses (time between contiguous clicks), burstiness (rate of change between a series of contiguous clicks), time between revisits. Figure 1 illustrates the time-of-day accesses of 3 users and Figure 2 illustrates the genre of the webpages accesses.

The clickstream data is parsed into "sessions" where a session is defined as a series of consecutive clicks delimited by pauses greater than 30 minutes as in (Kumar and Tomkins 2010). The number of sessions for our users varied from 42 to 205. The length of a session averaged from 14 to 130 pageviews. User sessions are the data points in our study of Web behavior.

3.2 Recurrent Neural Networks

Recurrent neural networks are characterized by symmetric connections between the neurons of a neural net where the output of one neuron can in turn affect its source neuron and where the weight of this connection is the same in both directions. There has been a resurgence in recurrent neural networks due to deep learning neural networks which stacks several recurrent neural networks, typically called *autoencoders*. Restricted Boltzmann machines (Hinton, Osindero, and Teh 2006) are a popular autoencoder used for deep learning. In general, it can be shown that reconstructing the

[1] http://www.diffbot.com

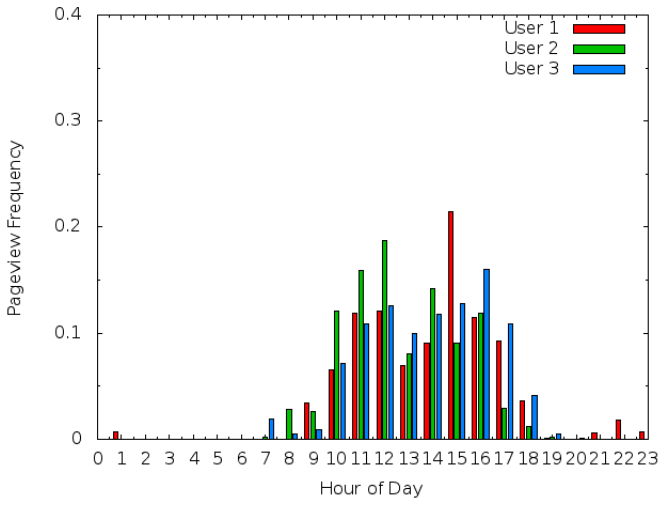

Figure 1: Time-of-day accesses for 3 users

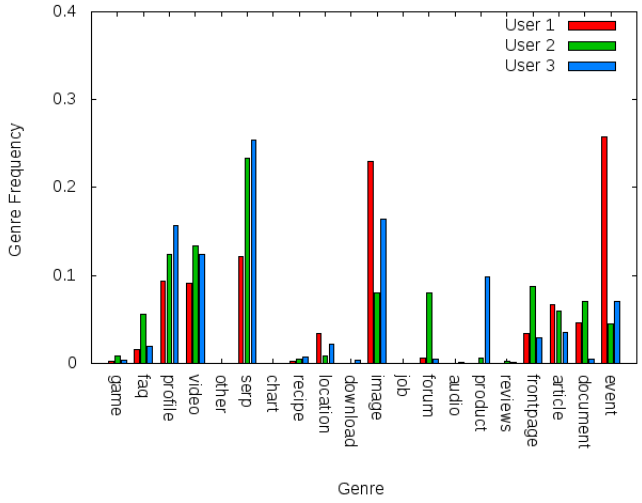

Figure 2: Genre accesses for 3 users

data through a generative process or through dimensionality reduction (for example, through principal component analysis) in an unsupervised learning step can enhance classification with discriminative learners.

3.3 Hopfield Networks

Hopfield neural networks (HNNs) are one of the oldest and most primitive recurrent neural networks (Hopfield 1982). HNNs address the problem of content-addressable memories where partial inputs can retrieve associated memories. For example, flavors can evoke memories:

> "And as soon as I had recognized the taste of the piece of madeleine soaked in her decoction of lime-blossom which my aunt used to give me ... immediately the old grey house upon the street, where her room was, rose up like a stage set to attach itself to the little pavilion opening on to the garden which had been built out behind it for my parents ...; and with the house the town, from morning to night and in all weathers, the Square where I used to be sent before lunch, the streets along which I used to run errands, the country roads we took when it was fine ..." (Proust 2006)

We simply follow the original HNN algorithm outlined in (Hopfield 1982) where the weights W of the neural networks are constructed by adding each memory V^s, or binary training vector, sequentially as follows:

$$W_{ij} = \sum_s (2V_i^s - 1)(2V_j^s - 1)$$

and where $W_{ii} = 0$. Given a binary testing vector, $V^{s'}$, the output of the neural network $O_i^{s'} = H[\sum_j W_{ij} V_j^{s'}]$ where H is the hard threshold activation function:

$$H = \begin{cases} 1 & if\, O_j^{s'} \geq 0 \\ -1 & otherwise \end{cases}$$

Bipolar inputs and outputs are used for testing and reconstruction since $H(\sum_j W_{ij} V_j^{s'}) = (2V_i^s - 1)$ if and only if $(2V_j^s - 1)V_j^{s'} = 1$.The update $V_i^{s'} = O_i^{s'}$ can be synchronous or asynchronous depending on whether all neurons are updated simultaneously or at different intervals. Using synchronous update, only one simultaneous time step is usually needed before the neural network reaches a stable pattern. Using asynchronous update, the network will reach a stable state after a series of updates. Unlike other neural networks, the weights are not modified and therefore training is fast. In Hopfield's model, a memory is found as the local minima to the energy function associated with the input vector in the network $E = -\frac{1}{2} \sum \sum_{i \neq j} W_{ij} V_i V_j$ since W is symmetric. The Hopfield model has been shown to always converge to a minimal energy. In our approach, the reconstructed vector is then compared to the training vectors with the Hamming distance and the closest one is selected as the output vector in our implementation.

3.4 Multiclass Classification

HNNs have limited recall capacity. It is helpful to encode memories with maximal separable distance to improve the recall rate and there has been some work in coding theory to ensure the orthogonality of the stored patterns. (Wang and Wang 2008) In this work, we investigate the application of machine learning techniques to combine individual binary learners that distinguish between two classes for multiclass classification in the retrieval of stored patterns in HNNs. Multiclass classification is formally defined as the problem of finding a classifier $c : X \rightarrow Y$ where Y is a set of labels of size $k \geq 3$ and X is the set of examples.

Tournament Approach A common method for reducing multiclass classification to binary classification is the one-versus-all method where one class is tested against all other classes and the class with the greatest accuracy against the other classes is selected. In this approach, care must be taken to balance the class examples in the training set, for

example by randomly sampling the examples of the other classes. Another method, the filter tree algorithm (Beygelzimer, Langford, and Ravikumar 2007), consists of fixing a binary tree to run binary tournaments between examples of each class at the first level (from the bottom) and then merge the winner examples (those for which the prediction was correct) to the second level of the tree (from the bottom) consisting of matches between four classes grouped into two sets, one set for each subtree. This process continues until the root node contains a classifier grouping all the classes into two groups. This algorithm effectively filters noisy and ambiguous examples from the training set. At test time, the test example cascades through the classifiers starting from the root until one of the binary classifiers at the leaves is reached. The time complexity of this algorithm is $log_2 n$ at test time where n is the number of classes. This algorithm is not however applicable to *class examplars* where there is effectively one example per class or where a set of instances constitute a concept without a label abstraction as can be found in our data. For example, the image of a specific person constitutes an examplar.

In our proposed tournament approach for HNNs, training and test are combined in a lazy approach. The process starts at the leaves like the training algorithm for filter trees. HNNs are constructed from training pairs and the test examplar is applied to each HNN to obtain a reconstructed examplar. This reconstructed examplar is then compared with the Hamming distance to the candidate examplars used to train the HNNs to determine the winner of the round. Figure 3 illustrates the process where a test examplar has been determined to be most similar to training examplar 4. However, at the next level of the tree (from the leaves), HNNs are formed dynamically depending on the winner examplars of tournament pairs (determined by the closest Hamming distance from the reconstructed vector). Algorithm 1 describes this recursive method. There are $n-1$ internal nodes in a binary tree of n leaves, so the time complexity of this algorithm is $O(n)$ and is better than the all-pairs classification. At most $\binom{n}{2}$ HNNs will be constructed and those can be cached and retrieved when needed with memoization.

All-pairs classification The all-pairs algorithm (Hastie and Tibshirani 1998) learns by training a classifier for each pair of classes. At test time, all $\binom{n}{2}$ classifiers are used to predict a class and the class with the most wins is selected with ties broken arbitrarily. No clear difference has been found empirically between the all-pairs algorithm and the filter tree algorithm on several datasets. (Beygelzimer, Langford, and Ravikumar 2007) In our proposed approach for HNNs, the networks for each pair are constructed once and applied to the test sets. Consequently, the time complexity of this algorithm is $O(n^2)$ at test time.

4 Empirical Results

We show empirical results on two different datasets, one obtained through our user study and one obtained through comScore [2], an internet data provider. We first describe our procedure for the user study dataset and then repeat the procedure for the comScore dataset. A summary of the results is described in Table 1.

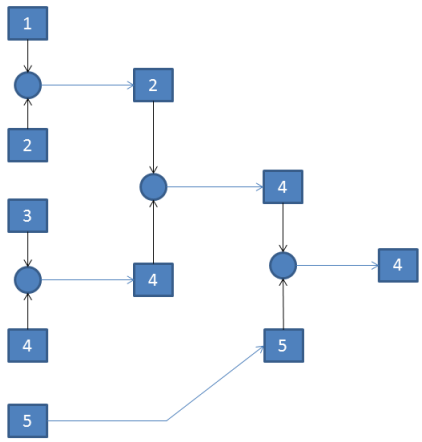

Figure 3: HNNs proposed tournament approach. The round nodes are the HNNs while the square nodes represent the class examplars.

Algorithm 1 HNNs Tournament recursive algorithm where partition is a function that breaks up a list into a sequence of disjoint lists of a given size (2).

```
MULTICLASS-TOURNAMENT (labels,test)
partitions ← partition (labels, 2)
bye ←labels\⋃partitions
winners ← winners⋃{bye}
FOR match ∈ partitions
  winner ← play-tournament(match,test)
  winners ← winners⋃{winner}
IF |winners| > 1
  MULTICLASS-TOURNAMENT(winners,test)
ELSE
  RETURN winners_0
```

For our experiments we extracted patterns of Web browsing behavior by genres and time-of-day accesses (Figure 4) from clickstream data obtained through our user study. The data is split evenly in a training and test sets according to user sessions. Extracting random clicks from the dataset produces identical distributions in the training and test sets. In that case, we obtain 100% accuracy for first place on the identification task for the 12 subjects in our user study using a simple Hamming distance in a nearest-neighbor approach. Splitting the dataset into temporally contiguous training and testing sets according to sessions gives potentially differ-

[2] http://www.comscore.com

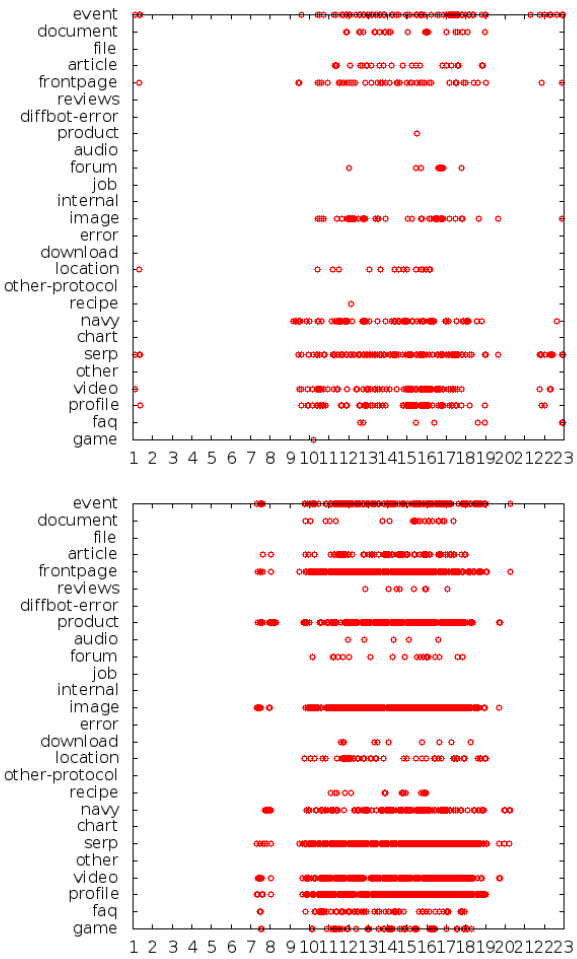

Figure 4: Two different user patterns of Web browsing behavior by genre and time-of-day. We can see that the user on the left works late at night while the user on the right starts working before commuting to work or stops to have breakfast.

ent distributions for the training and test sets and is a much harder prediction problem. In this case we obtain 75% accuracy for first place and 100% accuracy in the top 2 using the Hamming distance metric. We obtain the same results with the all-pairs algorithm while the tournament approach evaluated to 75% for first place and 83% for the top 2. A "bagging" approach (Breiman 1996) to sample the training set (with replacement) before applying the Hamming distance approach is another way to remove noise in the data. The encoding of the data into bipolar vectors loses the representation of multiple accesses of a genre on the same time-of-day but the bagging of data captures those regular browsing patterns while ignoring single visits. We extracted 1000 samples from the training set in our user study but this approach did not produce better results (averaged over 10 trials).

We scaled up our experiments on a dataset obtained from comScore consisting of 47 users and 8 weeks of data. The

Methods	**User Study**		**comScore**	
	1st	Top 2	1st	Top 2
Tournament	75	83	72	75
All-pairs	75	100	**73**	**81**
Hamming	**75**	**100**	72	79
Bagging	61±0.05	72±0.07	57±0.03	71±0.02

Table 1: Summary of results (%) on the different datasets temporally split into training and test sets according to sessions.

methodology used to capture this clickstream data was not known. This dataset is fairly noisy due to the high error rate from Diffbot in encoding the webpages into genres and the authenticity of the clicks (some automated clicks, such as Ajax requests, seem to have been included). The dataset was evenly temporally split into a training and test sets according to user sessions. The Hamming distance nearest-neighbor approach comparison gives 72% accuracy for first place and 79% accuracy in the top 2. The all-pairs approach gives 73% accuracy for first place and 81% accuracy in the top 2 which is slightly better but not significantly better. The "bagging" approach on the training set (2000 samples) did not produce significantly different results.

5 Conclusion

Representing Web browsing behavior as a spatial-temporal process of genres by time-of-day was shown to be sufficient to uniquely characterize users with a certain degree of accuracy. HNNs are very fast to implement and test. We have used multiclass classification techniques, tournament and all-pairs, to enhance their recall property with divide-and-conquer approaches suitable for parallelization. While we have not found a significantly improvement from our multiclass classification algorithms over the Hamming distance metric for identification based on the same bipolar representation of the data as HNNs, the all-pairs algorithm gave robust and consistent results albeit with slower run time than thr tournament recursive algorithm. Restricted Boltzmann machines (Hinton 2010) promise to further enhance the recall property of associative networks but were found slower to execute at this time. Further work will enhance this representation with a more accurate genre "palette" for identification. We will also look at further enhancing the recall and accuracy of associative neural networks of Web browsing behavior and adapt our algorithms for authentication.

References

Abramson, M., and Aha, D. W. 2013. User authentication from web browsing behavior. In *Florida Artificial Intelligence Society FLAIRS-26*.

Armstrong, R.; Freitag, D.; Joachims, T.; and Mitchell, T. 1995. Webwatcher: A learning apprentice for the world wide web. In *AAAI Spring Symposium on Information Gathering from Heterogeneous, distributed environments*.

Atterer, R.; Wnuk, M.; and Schmidt, A. 2006. Knowing the user's every move: user activity tracking for website us-

ability evaluation and implicit interaction. In *Proceedings of the International World Wide Web Conference WWW06*, 203–212. ACM.

Beygelzimer, A.; Langford, J.; and Ravikumar, P. 2007. Multiclass classification with filter trees. *Preprint, June* 2.

Breiman, L. 1996. Bagging predictors. *Machine learning* 24(2):123–140.

De Bock, K., and Van den Poel, D. 2010. Predicting website audience demographics for Web advertising targeting using multi-website clickstream data. *Fundamenta Informaticae* 98(1):49–70.

Hastie, T., and Tibshirani, R. 1998. Classification by pairwise coupling. *The annals of statistics* 26(2):451–471.

Hinton, G. E.; Osindero, S.; and Teh, Y.-W. 2006. A fast learning algorithm for deep belief nets. *Neural computation* 18(7):1527–1554.

Hinton, G. 2010. A practical guide to training restricted boltzmann machines. *Momentum* 9(1).

Hopfield, J. J. 1982. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences* 79(8):2554–2558.

Jain, A.; Bolle, R.; and Pankanti, S. 1999. *Biometrics: personal identification in networked society*. kluwer academic publishers.

Kumar, R., and Tomkins, A. 2010. A characterization of online browsing behavior. In *Proceedings of the 19th international conference on World wide web*, WWW '10, 561–570. New York, NY, USA: ACM.

Mills, D. 2011. Introducing browserid: a better way to sign in. Retrieved from http://identity.mozilla.com/post/7616727542/introducing-browserid-a-better-way-to-sign-in.

Narayanan, A.; Paskov, H.; Gong, N.; Bethencourt, J.; Stefanov, E.; Shin, E.; and Song, D. 2012. On the feasibility of internet-scale author identification. In *Proceedings of the 33rd Conference on IEEE Symposium on Security and Privacy*.

Padmanabhan, B., and Yang, C. 2006. Clickprints on the web: Are there signatures in web browsing data? Technical report, Wharton School, University of Pennsylvania.

Pentland, A., and Pentland, S. 2008. *Honest signals: how they shape our world*. The MIT Press.

Proust, M. 2006. *Remembrance of Things Past: Cities of the plain, The Captive, The Sweet cheat gone, Time regained. vol. two*, volume 2. Wordsworth Editions.

Salem, M., and Stolfo, S. 2010. Detecting masqueraders: A comparison of one-class bag-of-words user behavior modeling techniques. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications* 1(1):3–13.

Sweeney, L. 1996. Replacing personally-identifying information in medical records, the scrub system. In *Proceedings of the AMIA Annual Fall Symposium*, 333. American Medical Informatics Association.

Thibeau, D., and Reed, D. 2009. Open trust frameworks for open government. Retrieved from http://openid.net/government/Open_Trust_Frameworks_for_Govts.pdf.

Turkle, S. 2012. *Alone together: Why we expect more from technology and less from each other*. Basic Books.

Wang, S., and Wang, H. 2008. Password authentication using hopfield neural networks. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 38(2):265–268.

Wondracek, G.; Holz, T.; Kirda, E.; and Kruegel, C. 2010. A practical attack to de-anonymize social network users. In *IEEE Security and Privacy*.