A Dataset for Active Linguistic Authentication

Patrick Juola
Juola & Associates
pjuola@juolaassoc.com

John I. Noecker Jr.
Juola & Associates
jnoecker@juolaassoc.com

Ariel Stolerman
PSAL, Drexel University
ams573@cs.drexel.edu

Michael V. Ryan
Juola & Associates
mryan@juolaassoc.com

Patrick Brennan
Juola & Associates
pbrennan@juolaassoc.com

Rachel Greenstadt
PSAL, Drexel University
greenie@cs.drexel.edu

Abstract

Biometric technologies provide the possibility of a new and more effective way of security computers against unauthorized access; linguistic technologies, and in particular, authorship attribution technologies, provide the possibility of a means to this end. We report on a novel corpus developed to test this possibility. Using temporary workers in a simulated office environment, we collected a week's work-product for 19 subjects and demonstrate that techniques culled from the field of authorship attribution can identify workers with more than 90% accuracy.

1 Introduction

Standard password-based identification systems are well-known to have flaws. Passwords can be forgotten, written down, stolen, and guessed. If any of these events happen, an intruder has the keys to the kingdom. Recognizing this, biometric-based identification systems have been proposed that avoid or mitigate some of these issues. (It's hard to forget your own thumbprint.) But developing and testing these

systems can be a challenge precisely because of the need for a wide variety of humans, especially when the biometric task is challenging or time-consuming.

One possibility for biometric validation is the individual use of language. Prior work has shown that the authorship of documents as small as a few hundred words can be correctly identified. In a typical office environment, a worker will type many more words and thus potentially continually identify themselves to a suitable program over the course of a day. We report here on the development of a special purpose corpus to enable this type of analysis, and of some preliminary analyses of this corpus as a proof of concept.

2 Background

2.1 Authentication

Traditionally, the method to prevent someone from using your computer involves a password. Whether at startup time or through a lock screen, every user is given a (public) user name and a (private) password that must be typed to authenticate him or her. The flaws associated with this paradigm are equally traditional—passwords can be forgotten, which prevents access; to prevent forgetting, passwords will be written down and can be stolen, which enables unauthorized access; passwords can be guessed or cryptanalytically recovered, and so forth.

More subtly, traditional password-based protection is limited in the sort of protection it provides. Once a person presents their password, they are are authenticated. If I need to step away from my desk for a moment, anyone could step up to the keyboard and work their evil will on the system Chaski (Chaski, 2005) defined this as the "who's at the keyboard" dilemma and cites several examples, including the case of a dead body found in a room with a disputed suicide note typed on a computer, and the case of an email sent while an employee was away from her desk at lunch. Coulthard (Coulthard, mingb) describes a similar case involving a disputed email. In other words, traditional password protection provides only a passive, "perimeter" security that does not prevent "insiders" from abusing their status.

An improved security model would involve continuous, active authentication where the behavior of the person at the keyboard is monitored and security measures can be triggered immediately at any time when the behavior of the person changes. These measures can be as simple as re-authentication or as complicated as triggering a call to building security while locking the computer down.

2.2 Stylometry and Authorship Attribution

Authorship attribution, also called stylometry or stylistics, is a well-established (Juola, 2006; Koppel et al., 2009; Stamatatos, 2009; Jockers and Witten, 2010) field of study, although it has not typically been used for authentication before. The theory behind this field is that everyone has their own unique "stylome" (van Halteren et al., 2005), a unique set of idiolectal choices that describe their speaking and writing style. At a group level, this is the kind of choice that causes Brits to walk on "pavements" instead of "sidewalks," and at an individual level can be the kind of choice that causes you to place a fork "to" the left of the plate instead of "on" the left or "at" the left. Quantifying these choices, for example, by making a histogram of function words (Mosteller and Wallace, 1964; Binongo, 2003) or of character n-grams (Stamatatos, ming) enables investigators to develop a computationally tractable summary of stylistic choices and form judgements on based on these summaries.

Standard practice for stylometric investigations involves a detailed comparison of stylistic features culled from a training set of documents. The questioned document is then compared against the training set, typically using some form of classification or machine learning algorithm, and an appropriate decision taken. A related problem is authorship verification, where a novel document is compared to a summary of a single author; if the novel document is close (stylistically) to the summary, it is inferred to be by that author and if distant, not.

A third application of stylometric technology is in stylistic profiling (Koppel et al., 2005; Argamon et al., 2009; van Halteren, 2007; Gray and Juola, 2011; Juola et al., 2011), where the objective is not to identify a specific person, but instead to identify characteristics of the writer, such as age, gender, social class, native language, and so forth. Again, a typical study involves collecting samples of (e.g.) male writing and female writing, and then comparing a questioned document against the stylistic summaries to classify the document as male- or female-written.

The application of this technology to authentication is fairly straightforward. Instead of using a training set of documents, one uses a pseudo-document containing the user's long-term behavior, and "verifies" that the recent behavior at the keyboard is

consistent with this long-term behavior. A significant inconsistency of course would trigger a security response. We are therefore proposing the use of linguistic stylistics as a sort of biometric, the same thing that would be involved in the use of typing speed or mouse movements (Zheng et al., 2011).

As a purely text classification technology (i.e. not real-time, not using keystrokes and instead using finished documents, and often involving forced-choice comparisons between a fixed group of authors), authorship attribution is in some regards a mature technology. For example, at the recent PAN-2012 conference[1], the top three methods all classified more than 80% of 241 documents correctly with (in some cases) more than a dozen distractor authors. We have reason to believe that ultimately authorship *authentication* may be more accurate than this, as issues like automatic spelling correction [c.f. (Coulthard, mingb)] will not normalize individual writing patterns — someone who is a poor typist or who continually misspells "*toutch" (Wellman, 1936) will not have this idiosyncratic quirk airbrushed away. We expect that an appropriately chosen analytic method will eventually be able to achieve similar or better results in this novel context.

2.3 JGAAP

In light of the differences among possible analyses, an obvious question is "which method works best?" In order to address this question, the Evaluating Variations in Language laboratory at Duquesne University has developed a modular system for the development and comparative testing of authorship attribution methods.(Juola, 2006; Juola et al., 2009) This system, called JGAAP (Java Graphical Authorship Attribution Program) provides a large number of interchangeable analysis modules to handle different aspects of the analysis pipeline such as document preprocessing, feature selection, and analysis/visualization.

The JGAAP project has been very successful, creating one of the most well-known and widely used systems for authorship analysis, leading the way in the search for best practices, and developing a group of protocols accurate enough to have been used in court (Coulthard, minga). The work has generated dozens of papers in conferences and journals, including work by researchers at universities as far apart as Drexel, UCSD, and the Pedagogical University of Krakòw. Most importantly for the proposed work, it has created a standard, tested set of operational primitives (such as approximately two dozen ways to assess linguistic differences) (Stein and Argamon, 2006) based on different underlying cognitive models and computational approaches (Juola, 2002). We expect to be able to lever this toolset into a wide-ranging and systematic exploration of many different types of analysis and relationships. Taking combinatorics into account, the number of different ways to analyze a set of documents numbers in the millions and can be expanded by the inventive user with a moderate knowledge of computer programming. And it's freely available (from `www.jgaap.com`) making it a useful testbed for other studies outside the Duquesne.

For example, Grant (Grant, ming) describes a criminal case involving vocabulary comparisons among text messages sent by a number of people; the technical question involved in this case hinged on the existence and number of specific words (or spelling variants such as "wen" for "when" or "4get" for "forget") that were used by only one person involved. This can be captured by measuring document similarity using the so-called Jaccard or intersection distance, essentially a measure of vocabulary overlap without regard to specific frequencies. By contrast, the classic Mosteller-Wallace (Mosteller and Wallace, 1964) study of historical documents examined frequency differences among common (and therefore shared) vocabulary; the important question was not whether or not people used works like "upon" (because we all do), but whether the disputed document used that word more like person A or person B. This type of analysis can be done by measuring document similarity based on frequency measures such as normalized cosine aka dot-product distance (Noecker Jr. and Juola, 2009) or Manhattan distance. JGAAP has been expanded to include all three of these measures as well as many others.

We took advantage of JGAAP's expansiveness in one specific way for this experiment by developing a new preprocessor (Keylogger Canonicization) as detailed in a later section.

2.4 JStylo

One of JGAAP's progenies, JStylo [2] is an open-source authorship attribution platform developed in the Privacy, Security and Automation laboratory at Drexel University on top of the JGAAP project. The main

[1] `http://pan.webis.de`

[2] http://psal.cs.drexel.edu/

reason for its development is to allow cross-feature analysis, where multiple features can be extracted and included in one analysis, an option that was not available in JGAAP at the time. JStylo is developed as part of a dual analysis tool, along with Anonymouth (McDonald et al., 2012), a writing-style anonymization platform, whose underlying authorship attribution engine is JStylo.

In JStylo every feature can be one of two types: either a class of feature frequencies (e.g. the features "a", "b", ..., "z" for the "Letters" feature class) or some numeric evaluation of the input documents (e.g. Yule's Characteristic K). An additional advantage of JStylo is its fine resolution feature definition capabilities. Each feature is uniquely defined by a set of its own document-preprocessing tools, one unique feature extractor (the core of the feature), feature-postprocessing tools and normalization/factoring options. All of JGAAP core features are available in JStylo, in addition to some newly developed features (e.g. regular-expression-based extractors).

As for analyses capabilities, the main classification tools available in JStylo are from the popular data mining and machine learning platform Weka (Hall et al., 2009). Those include classifiers commonly used for authorship attribution, like support vector machines, neural networks, Naïve Bayes classifiers, decision trees etc. In addition, JStylo provides an implementation of the Writeprints authorship attribution technique, known for its high accuracy in scenarios with large number of authors (Abbasi and Chen, 2008).

Although JStylo lacks the maturity of JGAAP, it compensates with a vast range of features and more importantly the possibility to combine them. This capability is used in the preliminary analysis shown in this paper, where the extensive feature set used with the Writeprints method is applied to the collected data.

3 The Work Product Corpus

To gather a suitable corpus for validation in a work environment, we created a simulated work environment. A rented space in downtown Pittsburgh was used to create an office, staffed on a weekly basis by temporary employees. These employees, in turn, were supervised by Juola & Associates staffers and asked, over the course of the week, to research and write blog articles "related to Pittsburgh in some way," thus providing them with an incentive to use standard computer tools such as browsers and search engines to do the research and word processors to do the actual writing. This task was expected to take approximately six hours per day (except for a short initial day as described below, and was expected to provide a reasonable degree of topical similarity (so that people could not be trivially distinguished on the basis of the type of task they were doing) while allowing them enough freedom to be individually distinctive. In particular, employees were not restricted from accessing personal websites or playing standard games, "as long as the work gets done," and people were allowed to copy and paste as long as the final articles were the subjects' own work.

As might be expected, the most utilized applications on these computers were Internet Explorer and Microsoft Word.

Subjects were also advised that their computers were stuffed with tracking software, and in particular a macro recorder was used to measure keyboard use, including individual keystrokes as well as dynamic information such as timing, length of keypress, overlap between keys, and so forth. The macro recorder also measured mouse events including clicks and movements. Key logging software was used to record text as it was entered, including mapping text to specific applications, clipboard use, and browsing history. In light of this, employees were warned that, although good-faith effort would be made to "scrub" the data prior to analysis, all activity would be captured, including personal information such as Facebook account names and passwords and that it could not be guaranteed that 100% of such information would be removed. (People were given an opportunity to ask that specific strings such as passwords or user names be automatically redacted, but it may be easier just to change one's Facebook password or just not log on from work.)

In addition to the main tasks, there were two sorts of secondary tasks. During the first day of each week, the morning was spent in an orientation process that included the administration of a number of psychometric tests measuring traits like personality, self-esteem, and learning styles. (We do not report further in this paper on this aspect of the study.) During the final two hours of the day, employees were asked to perform a set of small, explicitly-defined tasks (microtasks) such as describing a specific local landmark or event or summarizing and article. This provides us with a set of very detailed, task-specific data collections with much tighter control, possibly creating a different environment for task-focused interindividual

comparisons.

Data collection is ongoing and by project end we hope to have at least 80 subjects.

4 Preliminary Analysis

4.1 Materials and methods

Our preliminary analysis is based on the first three weeks of data gathered according to the protocol described in the previous section. This data set comprised five days of work for each of 14 people (one participant had dropped out by failing to show up for work as hired).[3] This created approximately 280Mb of data, containing 17.5 million mouse and keyboard events and 23 thousand websites visited. This dataset is much richer than we used for the study reported here. We focus only on the language used in the keystroke events.

4.2 Daily Data

4.2.1 Methods

For the first phase of the preliminary analysis, we analyzed each day's worth of work as a unit, using hold-one-out cross-validation (in other words, each document was analyzed individually against the other 68 documents), in a rather content-agnostic way (using only character n-gram distribution frequencies). We recognize that requiring a full day's work prior to taking a security decision is not useful in practice but it provides a baseline against which smaller samples can be measured.

Our analysis was performed using JGAAP, using the canonicizers Unify Case (neutralizing all case distinctions), Normalize Whitespace (replacing tabs, newlines, and multiple spaces with a single space character), as well as the newly developed Keylogger canonicizer. This canonizer cleaned up the logs in several ways related to the specific aspects of key logs. For example, this classifier removed anything that didn't represent a keystroke. This included time/date stamps from the key logger, as well as information about what window a set of keystrokes came from, and also a bunch of whitespace to make the logs readable. [Note that this means that if they typed something in a browser, then switched to Word, there is nothing left in the log to tell you that. So you could have "google.com-ENTER-is a PittsburghHotels in Pittsburgh institution" (multiple window input mashed together)].

We then converted any special keys to single character representations. So -ENTER- above gets converted to a newline. Or arrow key -UP- gets converted to some placeholder that was unlikely to appear in the actual input.

Finally, analysis was performed using a simple nearest-neighbor classifier (each day was classified as the person who had produced the most similar other day's work product) using either Manhattan distance (aka L_1 distance) or intersection distance (aka Jaccard distance) on the basis of histograms of character n-grams of various lengths ranging from 1–5.

4.2.2 Results

The results of the preliminary experiments are attached as table 1. All results are ultimately out of 69 documents, but are reported out of number of definitive classifications (e.g. if two or more authors tied, that is reported as a lower number in the denominator in the table).

Analysis method	Results
Manhattan 1-grams	37 / 69
Manhattan 2-grams	53 / 69
Manhattan 3-grams	62 / 69
Manhattan 4-grams	58 / 69
Manhattan 5-grams	50 / 69
Intersection 1-gram	6 / 21
Intersection 3-gram	23 / 68
Intersection 4-gram	22 / 69
Intersection 5-gram	22 / 69

Table 1: Daily analysis classification results

4.2.3 Discussion

Based on these results, it is clear that individual people can be distinguished with high accuracy; our best result is 88.4% accurate (better, in fact, in purely nominal terms than the PAN-2012 winner). It is also clear that in this particular framework, Manhattan distance is a more promising and accurate measure than intersection distance, suggesting that it's more useful to measure frequency differences than mere presence/absence distinctions. Despite this, the fact that decisions were possible at all in more than 21 cases using individual characters and intersection distance hints at the power of using keyboard interactions as a forensic/security tool; in these 21 cases, there are demonstrably keys that some individuals

[3] One day's work for one participant was temporarily mislaid; this has since been fixed, but was not analyzed for this paper; hence we have only 69 days of work.

did not hit *at all* over the course of a day's work. We are therefore dealing with a much richer set of possible features and events than just alphanumeric characters and punctuation.

4.3 Fixed-Size Sliding Window

4.3.1 Methods

For the second phase of the preliminary analysis, we concatenated every user's keystrokes data together and redivided it into consecutive non-overlapping documents (windows) of some predefined fixed size, measured in words: 100, 500 and 1000. This type of analysis is closer to the active authentication problem we aim to solve, as any real-time monitoring system will eventually be based on evaluating sliding windows of user input on-the-fly, in attempts to catch unauthorized users. One of the challenges we face is to decrease the window size as much as possible (leading to a quicker response time), while keeping high accuracy, low false positives and false negatives (i.e. undetected unauthorized users and false alarms on authorized users, respectively).

As in the first phase, prior to analysis the data was stripped-down from any Keylogger metadata, special keys were converted to unique single-character placeholders and whitespaces were normalized. As opposed to before, the raw data was not case-unified, and ALL special keys (including -ENTER-, -TAB- etc.) were replaced with placeholders (rather than being converted to a newline, tab etc.) in order to preserve user typing characteristics as much as possible. Since the data includes representation of special characters, it is more accurate to say document lengths are measured in tokens, rather than words (e.g. ch$\beta\beta$Cch$\beta\beta$hicago, where β represents backspace).

Other than the construction of the dataset, the second main difference from the first phase analysis is the feature set. In this phase we use a close variation of the *Writeprints* (Abbasi and Chen, 2008) feature set, which includes a vast range of linguistic features across different levels of text. A summarized description of the features is presented in table 2. By using a rich linguistic feature set we are able to better capture the user's writing style. With the special-character placeholders, some features capture aspects of the user's style usually not found in a standard authorship problem settings. For instance, frequencies of backspaces and deletes provide some evaluation of the user's typo-rate (or lack of decisiveness).

Our analysis was performed in JStylo, using 10-fold cross-validation for evaluation. We used two Weka classifiers: the KNN classifier with $K = 1$ and Manhattan-distance (similar to the previous phase) and SMO SVM (Platt, 1998) with soft margin constant $C = 1$ and polynomial kernel with degree 1. SMO solves multi-class problems using pairwise binary classification. For both classifiers the features are normalized by default.

Group	Features
Lexical	Character count
	Avg. word-length
	Letters
	50 most common letter bigrams
	50 most common letter trigrams
	Percentage of letters
	Percentage of uppercase letters
	Percentage of digits
	Digits
	2-digit numbers
	3-digit numbers
	Word length distribution
	Special characters
Syntactic	50 most common function words
	Punctuation
	Part-of-speech (POS) tags
	50 most common POS bigrams
	50 most common POS trigrams
Content	50 most common words in the corpus
	50 most common word bigrams in the corpus
	50 most common word trigrams in the corpus
Idiosyncrasies	Common misspellings

Table 2: The *Writeprints*-inspired feature set.

4.3.2 Results

The results for the second phase preliminary experiments are shown in table 3.

4.3.3 Discussion

If it was clear from the first phase that individuals can be distinguished with high accuracy based on a day's worth of work units, these results show that in fact high distinguishability can be achieved by looking at token sequences of up to 1000 in length. Moreover, the statistically insignificant ($p < 0.01$) difference between the results for 500- and 1000-token windows implies verification may be achieved after

Window Size	Accuracy	Weighted Avg. FN	Weighted Avg. FP
SMO			
100	81.07%	18.93%	2%
500	93.06%	6.94%	0.9%
1000	93.33%	6.77%	0.9%
KNN			
100	71.79%	28.21%	2.4%
500	83.04%	16.96%	1.5%
1000	83.13%	16.87%	1.1%

Table 3: Sliding-window analysis classification results

merely 500 tokens are read from the user input. However, the statistically significant ($p < 0.01$) difference in accuracy when dropping to 100-token windows suggests that there is a minimal threshold to consider when using these settings.

In addition, support vector machines, which are used extensively before in authorship attribution problems due to their high performance and accuracy, prevailed in these settings as well and outperformed KNN ($p < 0.01$), with the best result of 93.33% accuracy for 1000-token windows (and for 500 not far behind with 93.06% accuracy).

5 Future Work and Conclusions

From a practical standpoint, this research at its current state (although perhaps not for incident response or post-hoc analysis); knowing at quarter to 5 that the person who has been at Harry's desk since early morning wasn't Harry will not prevent the intruder from doing his nefarious work all day. But these results nevertheless illustrate a successful prototype and proof of concept, and it will be easy enough in future work to use varying size windows or other types of smaller samples. We can, for instance, the accuracy achieved with temporal windows rather than length-based windows such as hour-long, five-minute long, or minute-long slices of work with this baseline.

Similarly, we have chosen only a few types of feature/event to analyze out of the dozens provided by the JGAAP and JStylo framework, and a similarly few types of classifiers (nearest-neighbor classification based on two different distance measures and SVM with out-of-the-box configuration) and ignored the possibility of using other classification techniques, including other distances or non-distance methods such as binary or even one-class support vector machines, neural networks, linear discriminant analysis, latent Dirichelet allocation, and so forth. In the longer run, it has also been shown that ensemble methods such as mixture-of-experts (Juola, 2008) tend to outperform individual analyses, and we need to investigate whether, if realistically small/short samples do not work well enough under single analysis, we can nevertheless achieve good authentication with multiple independent analyses. Similarly, we may be able to combine linguistic biometrics with other data sources [for example, with mouse movements, which have been shown to achieve good results (Zheng et al., 2011) as a base for authentication].

From a security standpoint, another key question is its accuracy in the face of active and deceptive hostility; put more simply, can I fake someone else's keyboard activities well enough to fool the security monitor. Again, the preliminary results do not take this into account, but recent work in stylistic deception such as (Brennan and Greenstadt, 2009; Juola and Vescovi, 2010; Juola and Vescovi, 2010; McDonald et al., 2012) provide a road map for further work.

Despite the limitations of the current analysis, we nevertheless feel that the results presented here show a promising proof-of-concept. The Department of Defense has suggested (DARPA BAA 12-06) that computer-captured biometrics can be "used to uniquely recognize humans" with high accuracy and minimal intrusiveness. We feel our work confirms this suggestion, providing approximately 90% accuracy across nearly two dozen experimental participants. While further work is required, both to improve accuracy and to address verification in unknown settings (adversaries not from the known set of users) — and possibly to integrate with other sources of biometric information and to develop a commercial-scale security system — the results here strongly confirm the promise of this approach.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. OCI-1032683 and by DARPA under BAA-12-06. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation or DARPA.

References

Abbasi, A. and Chen, H. (2008). Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace. *ACM Trans. Inf. Syst.*, 26(2):7:1–7:29.

Argamon, S., Koppel, M., Pennebaker, J. W., and Schler, J. (2009). Automatically profiling the author of an anonymous text. *CACM*, 52(2):119–123.

Binongo, J. N. G. (2003). Who wrote the 15th book of Oz? an application of multivariate analysis to authorship attribution. *Chance*, 16(2):9–17.

Brennan, M. and Greenstadt, R. (2009). Practical attacks against authorship recognition techniques. In *Proceedings of the Twenty-First Conference on Innovative Applications of Artificial Intelligence (IAAI)*, Pasadena, CA.

Chaski, C. E. (2005). Who's at the keyboard: Authorship attribution in digital evidence invesigations. *International Journal of Digital Evidence*, 4(1):n/a. Electronic-only journal: http://www.ijde.org, accessed 5.31.2007.

Coulthard, M. (Forthcominga). Authorship and immigration : A case study. *Brooklyn Law School Journal of Law and Policy*.

Coulthard, M. (Forthcomingb). On the admissibility of linguistic evidence. *Brooklyn Law School Journal of Law and Policy*.

Grant, T. (Forthcoming). Title not available at press time. *Brooklyn Law School Journal of Law and Policy*.

Gray, C. and Juola, P. (2011). Personality identification through on-line text analysis. In *Chicago Colloquium on Digital Humanities and Computer Science*, Chicago, IL.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18.

Jockers, M. L. and Witten, D. (2010). A comparative study of machine learning methods for authorship attribution. *LLC*, 25(2):215–23.

Juola, P. (2002). The significant of lexical choice in language change. In *Proceedings of Quantitative Investigation in Theoretical Linguistics (QITL)*, Osnabrueck, Germany.

Juola, P. (2006). Authorship attribution. *Foundations and Trends in Information Retrieval*, 1(3).

Juola, P. (2008). Authorship attribution : What mixture-of-experts says we don't yet know. In *Proceedings of American Association for Corpus Linguistics 2008*, Provo, UT USA.

Juola, P., Noecker, Jr., J., Ryan, M., and Speer, S. (2009). Jgaap 4.0 — a revised authorship attribution tool. In *Proceedings of Digital Humanities 2009*, College Park, MD.

Juola, P., Ryan, M., and Mehok, M. (2011). Geographically localizing tweets using stylometric analysis. In *Proceedings of the American Association of Corpus Linguistics 2011*, Atlanta, GA.

Juola, P. and Vescovi, D. (2010). Empirical evaluation of authorship obfuscation using JGAAP. In *Proceedings of the Third Workshop on Artificial Intelligence and Security*, Chicago, IL USA.

Koppel, M., Schler, J., and Argamon, S. (2009). Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology*, 60(1):9–26.

Koppel, M., Schler, J., and Zigdon, K. (2005). Determining an author's native language by mining a text for errors (short paper). In *Proceedings of KDD*, Chicago,IL.

McDonald, A. W. E., Afroz, S., Caliskan, A., Stolerman, A., and Greenstadt, R. (2012). Use fewer instances of the letter "i": Toward writing style anonymization. In *Lecture Notes in Computer Science*, volume 7384, pages 299–318. Springer.

Mosteller, F. and Wallace, D. L. (1964). *Inference and Disputed Authorship : The Federalist*. Addison-Wesley, Reading, MA.

Noecker Jr., J. and Juola, P. (2009). Cosine distance nearest-neighbor classification for authorship attribution. In *Proceedings of Digital Humanities 2009*, College Park, MD.

Platt, J. (1998). Fast training of support vector machines using sequential minimal optimization. In Schoelkopf, B., Burges, C., and Smola, A., editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press.

Stamatatos, E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, 60(3):538–56.

Stamatatos, E. (Forthcoming). Title not available at press time. *Brooklyn Law School Journal of Law and Policy*.

Stein, S. and Argamon, S. (2006). A mathematical explanation of Burrows' Delta. In *Proc. Digital Humanities 2006*, Paris, France.

van Halteren, H. (2007). Author verification by linguistic profiling: An exploration of the parameter space. *ACM Transactions on Speech and Language Processing*, 4:n/a.

van Halteren, H., Baayen, R. H., Tweedie, F., Haverkort, M., and Neijt, A. (2005). New machine learning methods demonstrate the existence of a human stylome. *Journal of Quantitative Linguistics*, 12(1):65–77.

Wellman, F. L. (1936). *The Art of Cross-Examination*. MacMillan, New York, 4th edition.

Zheng, N., Paloski, A., and Wang, H. (2011). An efficient user verification system via mouse movements. In *Proceedings of the 18th ACM conference on Computer and communications security*, CCS '11, pages 139–150, New York, NY, USA. ACM.