Dual-mode N-gram Similarity Detection for Forensic Authorship Analysis

John Blake
University of Aizu
Aizuwakamatsu
Japan
jblake@u-aizu.ac.jp

Kazuma Tamura
University of Aizu
Aizuwakamatsu
Japan
m5281051@u-aizu.ac.jp

Krzysztof Kredens
Aston University
Birmingham
United Kingdom
k.j.kredens@aston.ac.uk

Abstract

This paper introduces a dual-mode n-gram similarity detection tool specifically designed for corpus-based forensic authorship analysis. Intra-corpus mode is used to verify consistency within a dataset while inter-corpus mode is for comparison to a questioned dataset. Preliminary accuracy evaluation of shared n-gram detection is perfect at 100%.

1 Introduction

1.1 Background

Authorship analysis involves the attribution or exclusion of authorship based on linguistic evidence (Grant, 2013), a task that relies heavily on identifying patterns of lexical similarity and dissimilarity across documents. Forensic authorship analysis supports criminal and civil investigations by providing evidence-based attribution of anonymous or disputed texts (Coulthard and Johnson, 2000).

The datasets in forensic contexts tend to be rather small (Carter, 2022), frequently focussing on discovering the authorship of one text by comparing with known texts written by candidate authors. Linguistic similarity between questioned and known documents can offer probative value, especially when reinforced by recurrent and relatively rare (i.e. distinctive) lexical or syntactic patterns. Current approaches, however, rely on corpus query tools such as AntConc (Anthony, 2024), WordSmith Tools (Scott, 2008) and LancsBox (Brezina, 2025), which were developed for other purposes.

In a typical forensic authorship analysis workflow, linguists read and annotate texts to identify potentially distinctive n-grams (Wright, 2017). These n-grams may be examined in context using the keyword-in-context (KWIC) display in a corpus query tool to determine whether their usage is habitual or anomalous (Johnson and Wright, 2014).

When working with multiple documents assigned to either questioned (Q) or known (K) categories, it is essential to compare the n-gram distribution within each category. This intra-group or intra-corpus analysis helps assess authorial stylistic consistency (Cardaioli et al., 2021; Zhu and Jurgens, 2021). Subsequently, the analysis moves to the comparison of inter-group or inter-corpus shared n-grams, where distinctive n-grams in the Q text are checked for overlap in the K dataset, thus helping to establish or exclude authorship (Nini, 2018).

1.2 Problem

Current corpus tools are designed to prioritise frequency-based analysis of large datasets; they are not optimised for saliency (Boswijk and Coler, 2020) nor do they focus on fine-grained analysis of small datasets comprising one or more short texts. Yet, saliency and nuanced analysis of small datasets are of the utmost importance in forensic investigations, where the focus is on identifying potentially distinctive expressions that may distinguish and disambiguate authorship.

Forensic authorship analysts face two main difficulties. First, one way to confirm stylistic consistency within the texts attributed to one author is to discover how many distinctive n-grams are shared between the texts. To do so, the n-grams need to be identified, counted, ranked by frequency and compared among all the texts.

Second, once the distinctive n-grams have been identified, the distinctive n-grams that occur in both Q and K texts need to be compared, which involves identifying, counting, and ranking them by frequency of shared n-grams.

Presently, neither the consistency nor the comparison functionalities are directly available in any corpus tool. Thus, there is a niche that needs to be addressed to improve the workflow for forensic linguists adopting a shared n-gram approach.

1.3 Research Objectives

Our primary motivation is to bridge this gap by designing a user-friendly tool that consolidates these functions into a single platform. The two primary objectives of this project are as follows:

1. to integrate a consistency-checking function that enables users to assess whether documents attributed to a single author exhibit internal stylistic coherence; and

2. to facilitate comparison of shared n-grams between questioned texts and several candidate author datasets.

Together, these objectives form the foundation of the creation of a practical n-gram similarity tool for forensic authorship analysis.

1.4 Contribution

We present a dual-mode similarity detection function integrated into a web application. Unlike other corpus software, our tool supports cross-corpus alignment through an intuitive interface, enabling forensic linguists to:

- identify unigrams, bigrams, and trigrams occurring in both Q and K texts;

- assess stylistic consistency within a set of texts attributed to a single author; and

- compare shared n-grams across and between datasets.

By combining these capabilities, the tool offers a purpose-built solution for detecting potentially distinctive n-grams as indicators of authorship.

2 Related Work

2.1 Authorship Attribution and Verification

Authorship analysis has historically relied on stylometric features such as function word frequency, character n-grams, and syntactic structure (Ding et al., 2017; Klaussner et al., 2015; Lagutina et al., 2019). Statistical techniques include Burrows' Delta (Evert et al., 2017), support vector machines (Diederich et al., 2003), and nearest-neighbour classifiers (Cunningham and Delany, 2021). These methods have been used in authorship attribution in both literary contexts and forensic investigations.

These techniques involve a degree of familiarity with programming, which may range from simply adapting or running an existing program to creating a tailor-made solution. Added to this technical hurdle, in forensic contexts the use of sophisticated technologies should (or must in some jurisdictions) meet the evidentiary standards set out in the Daubert criteria (DeMatteo et al., 2019), which require that the methodology be scientifically valid, reliably applied, and open to scrutiny.

Explaining statistical models, mathematical reasoning, and computational procedures to a lay audience such as a jury presents a considerable challenge: the concepts are often abstract, highly technical, and removed from everyday experience (Coulthard, 2005). This complexity creates opportunities for opposing legal representatives to question, oversimplify, or misrepresent the underlying methods, potentially undermining the credibility of the expert witness testimony (Brodsky et al., 2012; O'Brien and O'Brien, 2017).

However, despite these technical advances, many forensic linguists still rely on workflows that combine multiple tools and require substantial manual intervention to extract, interpret, and triangulate stylistic patterns. Existing tools may be categorised as software libraries or ready-to-use tools.

2.2 Authorship Analysis Programs

Programs include Signature stylometric system[1], the well-respected stylometric R package Stylo (Eder et al., 2024) and the recently released R package Idiolect (Nini, 2024).

Signature provides a simple interface that is suitable for educational rather than forensic use. Stylo is designed for literary investigations of authorship rather than for forensic contexts and so relies on a stylometric approach. Idiolect builds on Nini's approach to linguistic individuality (Nini, 2023) and draws on the Likelihood Ratio Framework (Ishihara, 2021).

A commercial product NeoNeuro[2] offers authorship analysis via its rather dated proprietary program. There is no available details regarding its algorithm and effectiveness. NeoNeuro identifies 4-grams occurring in each of the K texts and compares them to the Q text. The output generated lists of the shared n-grams and provides percentage similiary score for each K text.

The Java Graphical Authorship Attribution Pro-

[1] https://www.philocomp.net/texts/signature.htm

[2] https://neoneuro.com/products/authorship-attribution

gram (JGAAP)[3] is straightforward to use because of its menu-driven user interface. This tool, however, needs to be set up, which may be difficult for those unfamiliar with GitHub.

Both Stylo and Idiolect require knowledge of R scripting, which is an onerous barrier for users with no programming experience. To set up JGAAP some programming knowledge is required although users can operate it without the need for any scripting. JGAAP, however, is no longer actively maintained, limiting its suitability for forensic work. Both Signature and NeoNeuro are simple to use, but very limited in terms of functionality.

General-purpose corpus analysis tools such as AntConc (Anthony, 2024), WordSmith Tools (Scott, 2008), LancsBox (Brezina, 2025), and Sketch Engine (Kilgarriff et al., 2014) were designed for linguistic research and teaching, not forensic investigation. AntConc excels at KWIC concordancing and keyword analysis but lacks built-in facilities for dataset comparison or intra-author stylistic consistency checks. Sketch Engine provides advanced collocational and profiling functions, but its emphasis on large-scale corpora does not align with the small, sensitive datasets typical of forensic work.

In practice, forensic linguists often adapt these tools through ad hoc workflows, combining outputs from multiple searches and using external software such as Excel or SPSS for comparison. This process is time-consuming, prone to error, and difficult to reproduce.

The absence of integrated cross-corpus and within-corpus comparison functionality in these tools means practitioners must either combine multiple outputs or write custom scripts to meet their needs.

2.3 Gap in Existing Approaches

The reviewed tools illustrate a clear gap: there is no single, forensic-oriented platform that integrates corpus management, exploration, and both intra- and inter-corpus similarity detection in a user-friendly environment. Our prototype addresses this gap by enabling forensic analysts to perform these tasks without programming skills, with an emphasis on clarity, reproducibility, and visual traceability.

3 System Overview

Figure 1 shows the system architecture of the corpus tool, focusing on the dual-mode similarity detection functionality. The system includes a file management module that enables users to perform standard Create, Read, Update and Delete (CRUD) operations, an analysis engine that includes the Similarity detection and comparison functionality, and a graphical user interface (GUI) to visualizer the results. The graphical user interface is divided into clearly labelled, colour-coded tabs, such as Manage, Search, Compare, Consistency, and Admin. Each tab corresponds to a major workflow step, allowing users to switch seamlessly between data upload, search configuration, cross-corpus comparison, intra-author consistency checks, and administrative controls. This modular structure supports intuitive navigation and ensures that forensic analysts can focus on linguistic patterns without being overwhelmed by interface complexity.

The system is implemented using a Django backend for robust server-side logic and data management, combined with a React-based single-page interface to ensure a responsive and intuitive user experience. Preprocessing of textual data is handled through the Natural Language Toolkit (NLTK) (Bird et al., 2009) in the development version, enabling tokenisation, normalisation, and n-gram extraction. The production release will harness SpaCy (Neumann et al., 2019), given its substantially faster processing speed, efficient memory management, and optimised pipeline for large-scale text handling.

Data exchange between the server and client is managed via JSON endpoints, ensuring efficient and lightweight communication. Corpora are stored as structured file sets with rich accompanying metadata, allowing for precise filtering and contextual retrieval. An indexing mechanism accelerates search and lookup operations, facilitating near-instantaneous access to relevant textual segments. The user interface organises functionality into clearly labelled tabs for corpus upload, keyword and pattern search, and similarity analysis. Within the similarity module, results are visually enhanced using colour-coded recurrence bars that indicate, at a glance, the number of documents in which a given n-gram occurs. This design balances technical performance with usability, allowing users to navigate between analytical functions

[3] https://github.com/evllabs/JGAAP

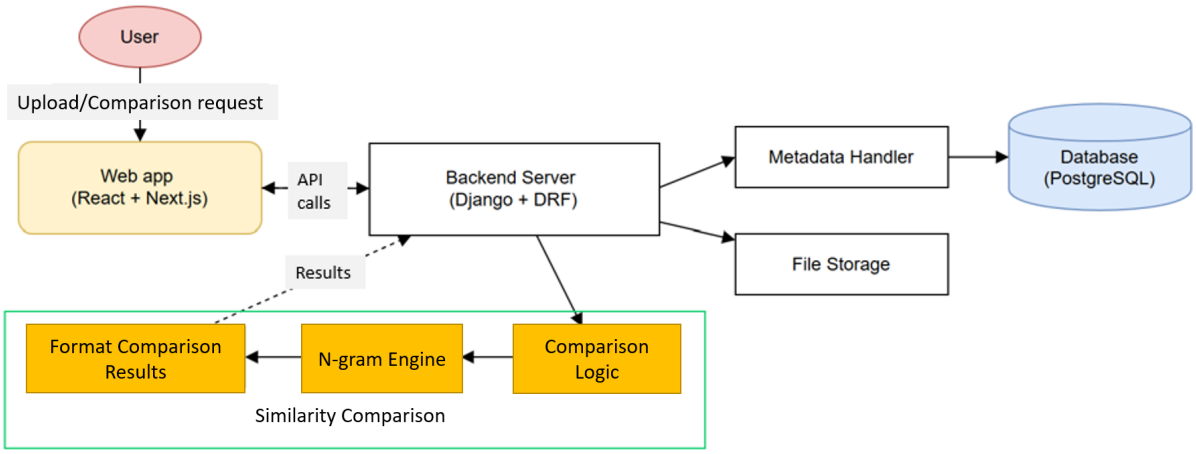

Figure 1: System architecture for similarity detection and comparison

without disrupting workflow.

4 Case Study: Threatening letters

A subset of the Threatening English Language (TEL) Corpus (Gales et al., 2023). was selected to show how the intra-corpus consistency and inter-corpus comparison functions can be harnessed with real-world forensic datasets.

To use the system, plain text files first need to be uploaded. Each file, regardless of its size or content, is treated as a single document. Metadata such as author ID and description can be added, and files are stored with secure identifiers to maintain confidentiality. Tokenization, part-of-speech tagging and basic preprocessing (e.g., lowercasing, punctuation removal) are automatically applied.

The KWIC interface allows users to search for any word, phrase, or regex pattern, retrieving and aligning all instances across the selected corpora as shown in Figure 2. Searches can also be performed using parts-of-speech tags.

Selecting the Compare tab provides access to the intra-corpus consistency and inter-corpus comparison features, both powered by the n-gram engine, which extracts unigrams, bigrams, and trigrams. The system calculates occurrence frequency and distribution across all selected documents, with results used in two modes.

The consistency checker (intra-corpus comparison) sorts results by the number of files in which shared n-grams occur. For any selected author corpus, n-gram recurrence is calculated across all documents, ranked by the number of matching files, and displayed in descending order from the most frequent n-grams (See Figure 3).

The cross-corpus comparison (inter-corpus comparison) sorts results by n-gram frequency in the Q corpus. One document or corpus is designated as Questioned (Q), and the system compares it against any number of Known (K) corpora, identifying overlapping n-grams. These are ranked by frequency in Q, with their frequency in each K corpus highlighted. Figure 4 shows the results of inter-corpus comparison of trigrams on the Compare tab using Searle 005 as the Questioned dataset (Q) and comparing that to two other known datasets, namely Searle 001 and Hickley 001. The first column gives the trigram in order of frequency in Q. The background of shared trigrams occurring in the other datasets is colourized. The raw count and the percentage of each trigram are also given.

5 Evaluation

The system was evaluated using multiple corpora: the Enron Email Corpus (Hussain, 2020), the Blog Authorship Corpus[4], the 100 Idiolects Project[5], and the Threatening English Language (TEL) Corpus (Gales et al., 2023).

In all cases, unigram, bigram, and trigram extraction ran successfully, and the comparison logic produced correct results. The system achieved 100% accuracy in counting, ranking, and comparing n-gram similarities both within and between corpora.

6 Discussion

The tool is transparent, easy to interpret, and requires no programming skills. Its highly visual,

[4] https://www.kaggle.com/datasets/rtatman/blog-authorship-corpus/data

[5] https://fold.aston.ac.uk/handle/123456789/17

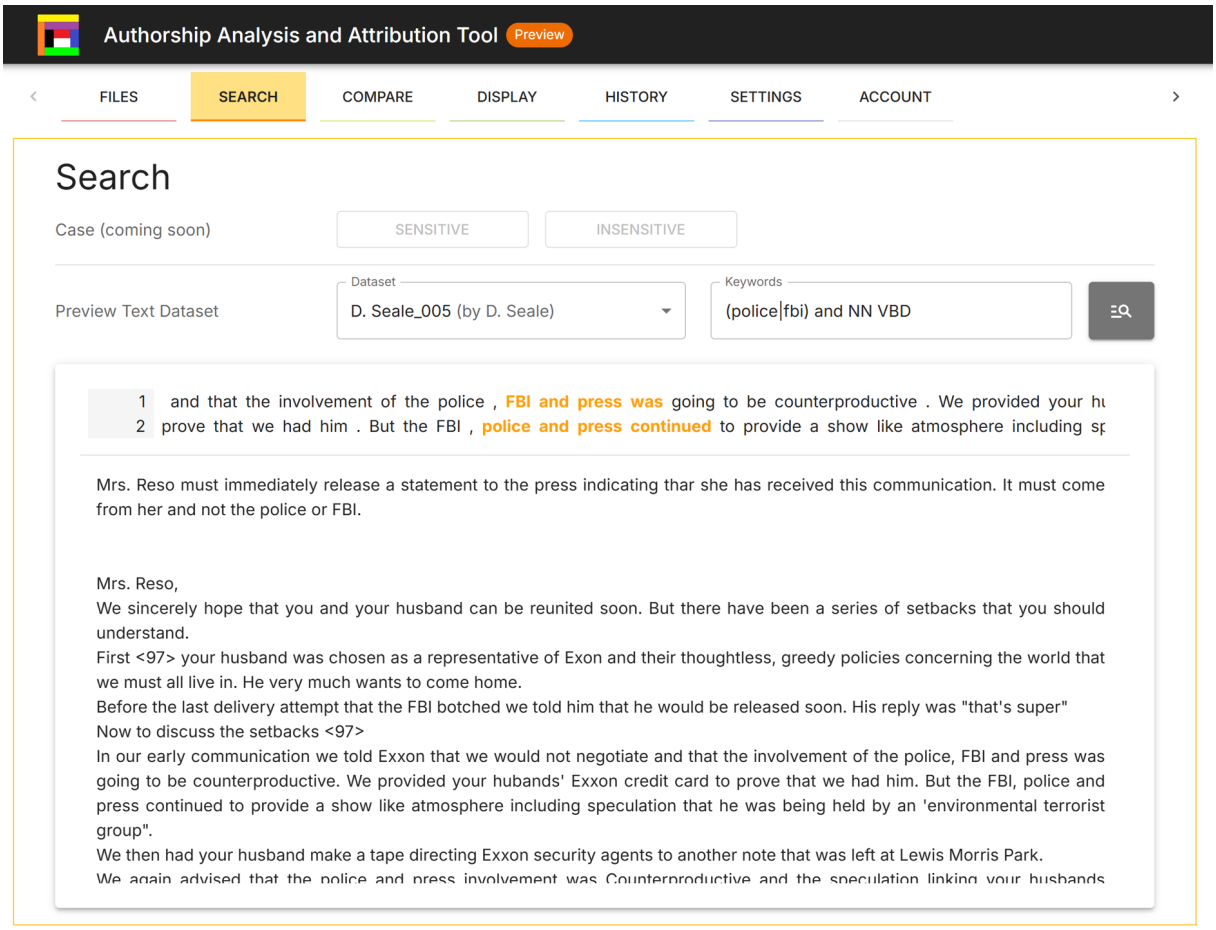

Figure 2: Screenshot of KWIC search.

intuitive interface enables non-technical users to become proficient with minimal training. The dual-mode n-gram similarity detection allows users to assess the consistency of n-gram usage within the texts of a single author (intra-corpus mode) and to evaluate similarity by comparing the questioned dataset with corpora from multiple authors (inter-corpus mode). Together, these capabilities streamline analysis, reduce reliance on multiple external tools, and support reproducible forensic workflows.

While the tool performs well for the intended tasks, several limitations remain. First, the current implementation is optimised for English texts, and performance on morphologically rich or low-resource languages has yet to be validated. Second, the accuracy of results depends on the quality of the input text. Errors introduced during transcription, OCR, or preprocessing may affect n-gram extraction and matching. Third, the system is designed for small to medium-sized datasets; although it can handle larger corpora, response times may increase, particularly during cross-corpus comparisons with multiple large K datasets.

7 Conclusion and Future Work

We have presented a dual-mode n-gram similarity detection tool purpose-built for forensic authorship analysis. The system addresses a niche not met by existing corpus or stylometric tools, enabling both intra-corpus stylistic consistency checks and inter-corpus comparison in a single, user-friendly environment. Its design emphasises transparency, reproducibility, and accessibility, making it suitable for forensic linguists without programming expertise. Evaluation on multiple datasets demonstrated perfect accuracy in shared n-gram detection and ranking.

We plan to release a production version with a two-tier access model and accompanying operational safeguards. The first tier will be a public demo environment offering read-only access to sandboxed corpora, with KWIC, frequency lists, and n-gram overlap on small sample datasets, capped query quotas, and limited file uploads, enabling immediate try-out without compromising data security. The second tier will provide pri-

Actions ANALYZE CONSISTENCY

Rows 50 | 1 2 3 4 5 ... 54

#	N-gram	Topology	Total Count	D. Seale_001 (D. Seale)	D. Seale_002 (D. Seale)	D. Seale_005 (D. Seale)	D. Seale_006 (D. Seale)	John W. Hinckley_001 (John W. Hickley)	Zodiac Killer_001 (Zodiac Killer)
1	warriors of the	4	4	1 (0.344%)	1 (0.202%)	1 (0.147%)	1 (0.248%)	0 (0.000%)	0 (0.000%)
2	of the rainbow	4	4	1 (0.344%)	1 (0.202%)	1 (0.147%)	1 (0.248%)	0 (0.000%)	0 (0.000%)
3	< 97 >	3	4	1 (0.344%)	1 (0.202%)	2 (0.293%)	0 (0.000%)	0 (0.000%)	0 (0.000%)
4	. if you	3	4	1 (0.344%)	0 (0.000%)	2 (0.293%)	0 (0.000%)	0 (0.000%)	1 (0.171%)
5	. we have	3	4	1 (0.344%)	1 (0.202%)	0 (0.000%)	2 (0.496%)	0 (0.000%)	0 (0.000%)
6	if you do	3	3	1 (0.344%)	0 (0.000%)	1 (0.147%)	0 (0.000%)	0 (0.000%)	1 (0.171%)
7	you do not	3	3	1 (0.344%)	0 (0.000%)	1 (0.147%)	0 (0.000%)	0 (0.000%)	1 (0.171%)
8	. warriors of	3	3	0 (0.000%)	1 (0.202%)	1 (0.147%)	1 (0.248%)	0 (0.000%)	0 (0.000%)
9	i 've got	2	4	0 (0.000%)	0 (0.000%)	0 (0.000%)	0 (0.000%)	1 (0.260%)	3 (0.514%)
10	place an ad	2	3	1 (0.344%)	0 (0.000%)	2 (0.293%)	0 (0.000%)	0 (0.000%)	0 (0.000%)
11	an ad in	2	3	1 (0.344%)	0 (0.000%)	2 (0.293%)	0 (0.000%)	0 (0.000%)	0 (0.000%)
12	ad in the	2	3	1 (0.344%)	0 (0.000%)	2 (0.293%)	0 (0.000%)	0 (0.000%)	0 (0.000%)

Figure 3: Screenshot of consistency function for trigrams.

vate workspaces for registered accounts with full access to all functionalities, including complete analysis history. The system will be deployed to a productive server using a containerised stack (e.g., Docker) with PostgreSQL, object storage for corpora, and a task queue for long-running jobs, supporting both single-tenant and multi-tenant configurations. Additional features will include single sign-on (SAML/OAuth2), encryption in transit and at rest, rate-limiting, audit logging, and monitoring

References

Laurence Anthony. 2024. Addressing the challenges of data-driven learning through corpus tool design–in conversation with laurence anthony. In *Corpora for language learning: Bridging the research-practice divide*, pages 9–18. Routledge.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc.

Vincent Boswijk and Matt Coler. 2020. What is salience? *Open Linguistics*, 6(1):713–722.

Vaclav Brezina. 2025. Corpus linguistics and ai:# lancsbox x in the context of emerging technologies. *International Journal of Language Studies*, 19(2).

Stanley L Brodsky, Caroline Titcomb, David M Sams, Kara Dickson, and Yves Benda. 2012. Hypothetical constructs, hypothetical questions, and the expert witness. *International Journal of Law and Psychiatry*, 35(5-6):354–361.

Matteo Cardaioli, Mauro Conti, Andrea Di Sorbo, Enrico Fabrizio, Sonia Laudanna, and Corrado A Visaggio. 2021. It's a matter of style: Detecting social bots through writing style consistency. In *2021 in-*

ternational conference on computer communications and networks (ICCCN), pages 1–9. IEEE.

Elisabeth Carter. 2022. Forensic linguistics. In *Handbook of Pragmatics*, pages 572–586. John Benjamins Publishing Company.

Malcolm Coulthard. 2005. The linguist as expert witness. *Linguistics and the Human Sciences*, 1(1):39–58.

Malcolm Coulthard and Alison Johnson. 2000. *Forensic Linguistics: An Introduction to Language in the Justice System*. Routledge.

Padraig Cunningham and Sarah Jane Delany. 2021. K-nearest neighbour classifiers-a tutorial. *ACM Computing Surveys (CSUR)*, 54(6):1–25.

David DeMatteo, Sarah Fishel, and Aislinn Tansey. 2019. Expert evidence: The (unfulfilled) promise of daubert. *Psychological Science in the Public Interest*, 20(3):129–134.

Joachim Diederich, Jörg Kindermann, Edda Leopold, and Gerhard Paass. 2003. Authorship attribution with support vector machines. *Applied intelligence*, 19(1):109–123.

Steven HH Ding, Benjamin CM Fung, Farkhund Iqbal, and William K Cheung. 2017. Learning stylometric representations for authorship analysis. *IEEE Transactions on Cybernetics*, 49(1):107–121.

Maciej Eder, Jan Rybicki, Mike Kestemont, Steffen Pielstroem, and Maintainer Maciej Eder. 2024. stylo: R package for stylometric analyses.

Stefan Evert, Thomas Proisl, Fotis Jannidis, Isabella Reger, Steffen Pielström, Christof Schöch, and Thorsten Vitt. 2017. Understanding and explaining delta measures for authorship attribution. *Digital Scholarship in the Humanities*, 32(suppl_2):ii4–ii16.

Tammy Gales, Andrea Nini, and Ellen Symonds. 2023. The threatening English language (TEL) corpus. https://research.manchester.ac.uk/en/datasets/the-threatening-english-language-tel-corpus. Accessed April 2025.

Tim Grant. 2013. TXT 4N6: Method, consistency, and distinctiveness in the analysis of SMS text messages. *Journal of Law and Policy*, 21(2):467–494.

Javed Hussain. 2020. Enron email dataset.

Shunichi Ishihara. 2021. Score-based likelihood ratios for linguistic text evidence with a bag-of-words model. *Forensic Science International*, 327:110980.

Alison Johnson and David Wright. 2014. Identifying idiolect in forensic authorship attribution: an n-gram textbite approach. *Language and Law/Linguagem e Direito*, 1(1).

Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlỳ, and Vít Suchomel. 2014. The Sketch Engine. *Lexicography*, 1(1):7–36.

Carmen Klaussner, John Nerbonne, and Çağrı Çöltekin. 2015. Finding characteristic features in stylometric analysis. *Digital Scholarship in the Humanities*, 30(suppl_1):i114–i129.

Ksenia Lagutina, Nadezhda Lagutina, Elena Boychuk, Inna Vorontsova, Elena Shliakhtina, Olga Belyaeva, Ilya Paramonov, and PG Demidov. 2019. A survey on stylometric text features. In *2019 25th Conference of Open Innovations Association (FRUCT)*, pages 184–195.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. Scispacy: Fast and robust models for biomedical natural language processing. *arXiv preprint arXiv:1902.07669*.

Andrea Nini. 2018. An authorship analysis of the jack the ripper letters. *Digital Scholarship in the Humanities*, 33(3):621–636.

Andrea Nini. 2023. *A theory of linguistic individuality for authorship analysis*. Cambridge University Press.

Andrea Nini. 2024. Idiolect: An R package for forensic authorship analysis.

Thomas C O'Brien and David D O'Brien. 2017. Effective strategies for cross-examining an expert witness. *Litigation*, 44(1):26–30.

Mike Scott. 2008. Developing wordsmith. *International Journal of English Studies*, 8(1):95–106.

David Wright. 2017. Using word n-grams to identify authors and idiolects: A corpus approach to a forensic linguistic problem. *International Journal of Corpus Linguistics*, 22(2):212–241.

Jian Zhu and David Jurgens. 2021. Idiosyncratic but not arbitrary: Learning idiolects in online registers reveals distinctive yet consistent individual styles. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 279–297, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Actions CHECK COMPARISON

Rows 50 1 2 3 4 5 ... 14

#	Trigram	Q: D. Seale_005 (D. Seale)	K1: D. Seale_001 (D. Seale)	K2: John W. Hinckley_001 (John W. Hickley)
1	a series of	2 (0.293%)	0 (0.000%)	0 (0.000%)
2	< 97 >	2 (0.293%)	1 (0.344%)	0 (0.000%)
3	that the fbi	2 (0.293%)	0 (0.000%)	0 (0.000%)
4	that we would	2 (0.293%)	0 (0.000%)	0 (0.000%)
5	the police ,	2 (0.293%)	1 (0.344%)	0 (0.000%)
6	police , fbi	2 (0.293%)	1 (0.344%)	0 (0.000%)
7	, fbi and	2 (0.293%)	0 (0.000%)	0 (0.000%)
8	fbi and press	2 (0.293%)	0 (0.000%)	0 (0.000%)
9	police and press	2 (0.293%)	0 (0.000%)	0 (0.000%)
10	. we then	2 (0.293%)	0 (0.000%)	0 (0.000%)
11	to a phone	2 (0.293%)	0 (0.000%)	0 (0.000%)
12	they would have	2 (0.293%)	0 (0.000%)	0 (0.000%)
13	tape of your	2 (0.293%)	0 (0.000%)	0 (0.000%)
14	of your husband	2 (0.293%)	0 (0.000%)	0 (0.000%)
15	more concerned with	2 (0.293%)	0 (0.000%)	0 (0.000%)
16	concerned with apprehension	2 (0.293%)	0 (0.000%)	0 (0.000%)
17	with apprehension than	2 (0.293%)	0 (0.000%)	0 (0.000%)
18	apprehension than with	2 (0.293%)	0 (0.000%)	0 (0.000%)
19	than with your	2 (0.293%)	0 (0.000%)	0 (0.000%)
20	with your husbands	2 (0.293%)	0 (0.000%)	0 (0.000%)
21	place an ad	2 (0.293%)	1 (0.344%)	0 (0.000%)
22	an ad in	2 (0.293%)	1 (0.344%)	0 (0.000%)
23	ad in the	2 (0.293%)	1 (0.344%)	0 (0.000%)

Figure 4: Screenshot of comparison function for trigrams.