Authorship Verification

Ph.D. Thesis Defense

Ariel Stolerman

Advisor: Dr. Rachel Greenstadt

Department of Computer Science
Drexel University
stolerman@cs.drexel.edu

April 20, 2015

Outline

Outline

1. Introduction
2. Background
3. Native Language Identification
4. Active Authentication
5. *Classify-Verify*
6. Summary

Introduction

- Stylometry: The study of linguistic style
  - Applied to authorship attribution: *Who wrote this document?*
- Authorship Verification:
  - *Given a document D and an author A, was D written by A?*
- Why Verification?
  - confidence – how sure are we in the results?
  - Tunable rigidity – natural for open-world problems
  - Verification can improve classification

Introduction

- Stylometry: The study of linguistic style
  - Applied to authorship attribution: *Who wrote this document?*
- Authorship Verification:
  - *Given a document D and an author A, was D written by A?*
- Why Verification?
  - confidence – how sure are we in the results?
  - Tunable rigidity – natural for open-world problems
  - Verification can improve classification

Introduction

- Stylometry: The study of linguistic style
  - Applied to authorship attribution: *Who wrote this document?*
- Authorship Verification:
  - *Given a document D and an author A, was D written by A?*
- Why Verification?
  - confidence – how sure are we in the results?
  - Tunable rigidity – natural for open-world problems
  - Verification can improve classification

Introduction

- Stylometry: The study of linguistic style
  - Applied to authorship attribution: *Who wrote this document?*
- Authorship Verification:
  - *Given a document D and an author A, was D written by A?*
- Why Verification?
  - confidence – how sure are we in the results?
  - Tunable rigidity – natural for open-world problems
  - Verification can improve classification

Introduction

- Stylometry: The study of linguistic style
  - Applied to authorship attribution: *Who wrote this document?*
- Authorship Verification:
  - *Given a document D and an author A, was D written by A?*
- Why Verification?
  - confidence – how sure are we in the results?
  - Tunable rigidity – natural for open-world problems
  - Verification can improve classification

Introduction

- Stylometry: The study of linguistic style
  - Applied to authorship attribution: *Who wrote this document?*
- Authorship Verification:
  - *Given a document D and an author A, was D written by A?*
- Why Verification?
  - confidence – how sure are we in the results?
  - Tunable rigidity – natural for open-world problems
  - Verification can improve classification

Introduction

- Stylometry: The study of linguistic style
  - Applied to authorship attribution: *Who wrote this document?*
- Authorship Verification:
  - *Given a document D and an author A, was D written by A?*
- Why Verification?
  - confidence – how sure are we in the results?
  - Tunable rigidity – natural for open-world problems
  - Verification can improve classification

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS$^+$13, FSA$^+$13, JNS$^+$13, SFG$^+$14, FSA$^+$14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS+13, FSA+13, JNS+13, SFG+14, FSA+14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Introduction – Contd.

Authorship verification Research:

- Generalization & Problem Relaxation for Improved Classification
  - Classification granularity $\leftrightarrow$ accuracy & confidence
  - Generalize problem $\rightarrow$ improve original problem
  - *Native Language vs. Language Family Identification* [SCG13]
- Stylometry-Based Security Applications
  - High-level authentication & identification
  - *Active Authentication* [JNJS$^+$13, FSA$^+$13, JNS$^+$13, SFG$^+$14, FSA$^+$14]
- Open-world settings
  - The true author may be missing from the set of candidates
  - *The Classify-Verify Algorithm* [SOAG14, SG]

Outline

1. Introduction
2. Background
3. Native Language Identification
4. Active Authentication
5. *Classify-Verify*
6. Summary

Stylometry

- Authorship attribution using linguistic style learned from text
- Everyone has a "stylistic fingerprint"
- Domain dominated by AI methods
  - NLP for text quantification
  - Machine learning for classification
- Current state of supervised stylometry: pretty good!
- Authorship Verification: Did *A* write *D*?
  - Relatively unexplored
  - Extremely relevant for security & online domains

Stylometry

- Authorship attribution using linguistic style learned from text
- Everyone has a “stylistic fingerprint”
- Domain dominated by AI methods
  - NLP for text quantification
  - Machine learning for classification
- Current state of supervised stylometry: pretty good!
- Authorship Verification: Did *A* write *D*?
  - Relatively unexplored
  - Extremely relevant for security & online domains

Stylometry

- Authorship attribution using linguistic style learned from text
- Everyone has a “stylistic fingerprint”
- Domain dominated by AI methods
  - NLP for text quantification
  - Machine learning for classification
- Current state of supervised stylometry: pretty good!
- Authorship Verification: Did *A* write *D*?
  - Relatively unexplored
  - Extremely relevant for security & online domains

Stylometry

- Authorship attribution using linguistic style learned from text
- Everyone has a "stylistic fingerprint"
- Domain dominated by AI methods
  - NLP for text quantification
  - Machine learning for classification
- Current state of supervised stylometry: pretty good!
- Authorship Verification: Did *A* write *D*?
  - Relatively unexplored
  - Extremely relevant for security & online domains

Stylometry

- Authorship attribution using linguistic style learned from text
- Everyone has a “stylistic fingerprint”
- Domain dominated by AI methods
  - NLP for text quantification
  - Machine learning for classification
- Current state of supervised stylometry: pretty good!
- Authorship Verification: Did A write D?
  - Relatively unexplored
  - Extremely relevant for security & online domains

Stylometry

- Authorship attribution using linguistic style learned from text
- Everyone has a “stylistic fingerprint”
- Domain dominated by AI methods
  - NLP for text quantification
  - Machine learning for classification
- Current state of supervised stylometry: pretty good!
- Authorship Verification: Did A write D?
  - Relatively unexplored
  - Extremely relevant for security & online domains

Stylometry

- Authorship attribution using linguistic style learned from text
- Everyone has a "stylistic fingerprint"
- Domain dominated by AI methods
  - NLP for text quantification
  - Machine learning for classification
- Current state of supervised stylometry: pretty good!
- Authorship Verification: Did *A* write *D*?
  - Relatively unexplored
  - Extremely relevant for security & online domains

Domain Problems

- Document D, documents $\mathcal{D}$, author A, authors $\mathcal{A}$
- Problems:
  - Most common – closed-world, supervised: Who in $\mathcal{A}$ wrote D?
  - Unsupervised: Segment D (or $\mathcal{D}$) by authors
  - Verification: Is D written by A?
- Baseline for other problems: mixed open/closed-world stylometry, author profiling

Domain Problems

- Document D, documents $\mathcal{D}$, author A, authors $\mathcal{A}$
- Problems:
  - Most common – closed-world, supervised: Who in $\mathcal{A}$ wrote D?
  - Unsupervised: Segment D (or $\mathcal{D}$) by authors
  - Verification: Is D written by A?
- Baseline for other problems: mixed open/closed-world stylometry, author profiling

Domain Problems

- Document D, documents $\mathcal{D}$, author A, authors $\mathcal{A}$
- Problems:
  - Most common – closed-world, supervised: Who in $\mathcal{A}$ wrote D?
  - Unsupervised: Segment D (or $\mathcal{D}$) by authors
  - Verification: Is D written by A?
- Baseline for other problems: mixed open/closed-world stylometry, author profiling

Domain Problems

- Document D, documents $\mathcal{D}$, author A, authors $\mathcal{A}$
- Problems:
  - Most common – closed-world, supervised: Who in $\mathcal{A}$ wrote D?
  - Unsupervised: Segment D (or $\mathcal{D}$) by authors
  - Verification: Is D written by A?
- Baseline for other problems: mixed open/closed-world stylometry, author profiling

Domain Problems

- Document D, documents $\mathcal{D}$, author A, authors $\mathcal{A}$
- Problems:
  - Most common – closed-world, supervised: Who in $\mathcal{A}$ wrote D?
  - Unsupervised: Segment D (or $\mathcal{D}$) by authors
  - Verification: Is D written by A?
- Baseline for other problems: mixed open/closed-world stylometry, author profiling

JStylo: an Authorship Attribution Framework

- Open-source Java authorship attribution research platform [MAC+12]
  - Define problem → set features → set classifiers → analyze
- Used by Anonymouth for anonymizing documents
- Powered by JGAAP, Weka [Juo, HFH+09]

JStylo: an Authorship Attribution Framework

- Open-source Java authorship attribution research platform [MAC+12]
  - Define problem → set features → set classifiers → analyze
- Used by Anonymouth for anonymizing documents
- Powered by JGAAP, Weka [Juo, HFH+09]

JStylo: an Authorship Attribution Framework

- Open-source Java authorship attribution research platform [MAC+12]
  - Define problem → set features → set classifiers → analyze
- Used by Anonymouth for anonymizing documents
- Powered by JGAAP, Weka [Juo, HFH+09]

JStylo: an Authorship Attribution Framework

- Open-source Java authorship attribution research platform [MAC$^+$12]
  - Define problem $\rightarrow$ set features $\rightarrow$ set classifiers $\rightarrow$ analyze
- Used by Anonymouth for anonymizing documents
- Powered by JGAAP, Weka [Juo, HFH$^+$09]

Outline

Native Language Identification

- Generalization & problem relaxation with verification
- Definitions:
  - L1: native language
  - L2: non-native language
  - LF: language family
- Problem: *Given L2 text, what is the author's L1(s)?*
  - L1-L2 transfer effect $\rightarrow$ LF-L2 transfer effect?
  - Increase L1-ID via LF-ID?
    - Yes – with verification + generalization

Native Language Identification

- Generalization & problem relaxation with verification
- Definitions:
  - L1: native language
  - L2: non-native language
  - LF: language family
- Problem: *Given L2 text, what is the author's L1(s)?*
  - L1-L2 transfer effect $\rightarrow$ LF-L2 transfer effect?
  - Increase L1-ID via LF-ID?
    - Yes – with verification + generalization

Native Language Identification

- Generalization & problem relaxation with verification
- Definitions:
  - L1: native language
  - L2: non-native language
  - LF: language family
- Problem: *Given L2 text, what is the author's L1(s)?*
  - L1-L2 transfer effect → LF-L2 transfer effect?
  - Increase L1-ID via LF-ID?
    - Yes – with verification + generalization

Native Language Identification

- Generalization & problem relaxation with verification
- Definitions:
  - L1: native language
  - L2: non-native language
  - LF: language family
- Problem: *Given L2 text, what is the author's L1(s)?*
  - L1-L2 transfer effect $\rightarrow$ LF-L2 transfer effect?
  - Increase L1-ID via LF-ID?
    - Yes – with verification + generalization

Native Language Identification

- Generalization & problem relaxation with verification
- Definitions:
  - L1: native language
  - L2: non-native language
  - LF: language family
- Problem: *Given L2 text, what is the author's L1(s)?*
  - L1-L2 transfer effect $\rightarrow$ LF-L2 transfer effect?
  - Increase L1-ID via LF-ID?
    - Yes – with verification + generalization

Native Language Identification

- Generalization & problem relaxation with verification
- Definitions:
  - L1: native language
  - L2: non-native language
  - LF: language family
- Problem: *Given L2 text, what is the author's L1(s)?*
  - L1-L2 transfer effect $\rightarrow$ LF-L2 transfer effect?
  - Increase L1-ID via LF-ID?
    - Yes – with verification + generalization

Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

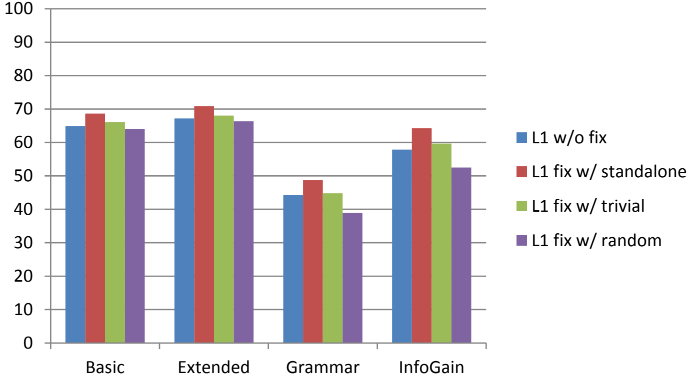
Native Language Identification – Method

- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Method

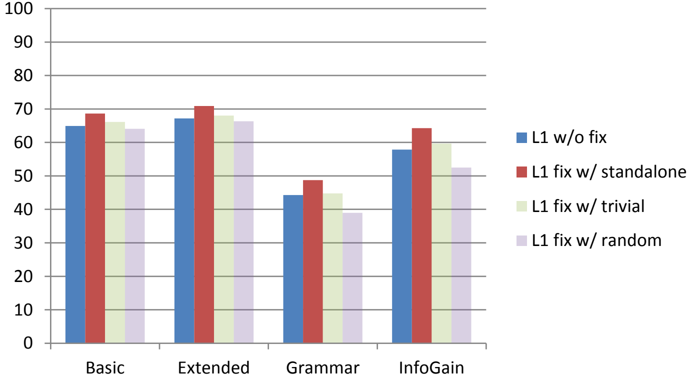
- Corpus: 11 L1s of 3 LFs from ICLEv2
- Features: 4 sets, using syntax and idiosyncrasies
- Classifier: SVM cross-validation, measured TPR
- Method – correct L1-ID by LF-ID:
  - Apply L1-ID, measure chosen L1 probability p
  - Set confidence threshold t
  - If $p \geq t$: take chosen L1
  - If $p < t$:
    - Apply LF-ID by Standalone / Trivial / Random
    - Reapply L1-ID only among languages in chosen LF
    - Take chosen L1

Native Language Identification – Eval



Native Language Identification – Eval

3.67%-6.43% increase in TPR using Standalone correction



Outline

Active Authentication

- Stylometry-based security application
- Active Authentication
  - The process of continuously verifying a user based on his/her ongoing interaction with the computer
- Problem: *Who is at the keyboard?*
  - Using real-time stylometric sensors
  - High-paced decision making
  - Natural for verification: doubting the user in front of us

Active Authentication

- Stylometry-based security application
- Active Authentication
  - The process of continuously verifying a user based on his/her ongoing interaction with the computer
- Problem: *Who is at the keyboard?*
  - Using real-time stylometric sensors
  - High-paced decision making
  - Natural for verification: doubting the user in front of us

Active Authentication

- Stylometry-based security application
- Active Authentication
  - The process of continuously verifying a user based on his/her ongoing interaction with the computer
- Problem: *Who is at the keyboard?*
  - Using real-time stylometric sensors
  - High-paced decision making
  - Natural for verification: doubting the user in front of us

Active Authentication

- Stylometry-based security application
- Active Authentication
  - The process of continuously verifying a user based on his/her ongoing interaction with the computer
- Problem: *Who is at the keyboard?*
  - Using real-time stylometric sensors
  - High-paced decision making
  - Natural for verification: doubting the user in front of us

Active Authentication

- Stylometry-based security application
- Active Authentication
  - The process of continuously verifying a user based on his/her ongoing interaction with the computer
- Problem: *Who is at the keyboard?*
  - Using real-time stylometric sensors
  - High-paced decision making
  - Natural for verification: doubting the user in front of us

Active Authentication

- Stylometry-based security application
- Active Authentication
  - The process of continuously verifying a user based on his/her ongoing interaction with the computer
- Problem: *Who is at the keyboard?*
  - Using real-time stylometric sensors
  - High-paced decision making
  - Natural for verification: doubting the user in front of us

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: `ch`$\beta\beta$`Cch`$\beta\beta$`hicago` $\Rightarrow$ `Chicago`
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: `ch`$\beta\beta$`Cch`$\beta\beta$`hicago` $\Rightarrow$ `Chicago`
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: $\texttt{ch}\beta\beta\texttt{Cch}\beta\beta\texttt{hicago} \Rightarrow \texttt{Chicago}$
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: ch$\beta\beta$Cch$\beta\beta$hicago $\Rightarrow$ Chicago
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: `ch`$\beta\beta$`Cch`$\beta\beta$`hicago` $\Rightarrow$ `Chicago`
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: `ch`$\beta\beta$`Cch`$\beta\beta$`hicago` $\Rightarrow$ `Chicago`
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: ch$\beta\beta$Cch$\beta\beta$hicago $\Rightarrow$ Chicago
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

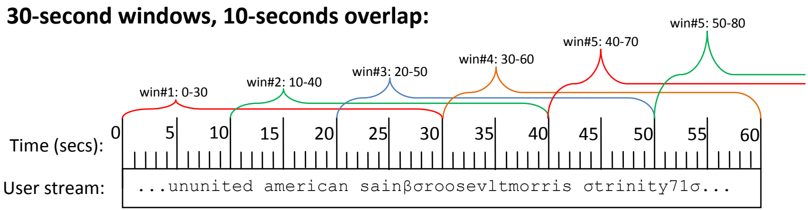
Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: $\texttt{ch}\beta\beta\texttt{Cch}\beta\beta\texttt{hicago} \Rightarrow \texttt{Chicago}$
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

Active Authentication – Method

- Corpus: Active Linguistic Authentication Dataset [JNJS+13]
- Features: variation of *Writeprints* [AC08]
  - Track special keys: backspace (β), shift (σ)...
  - Apply them: `ch`$\beta\beta$`Cch`$\beta\beta$`hicago` $\Rightarrow$ `Chicago`
- Classifier: SVM trained on 67 users
- Method
  - Initial day/#words-based windows, 14 users: 88–93% accuracy
  - Here: time-based overlapping sliding windows
    - Size (overlap): 10s, 30s, 60s (10s) & 5m, 10m, 20m (60s)
    - minimum characters/window: 100, 200, ..., 1000
  - Goal: use in multi-modal systems

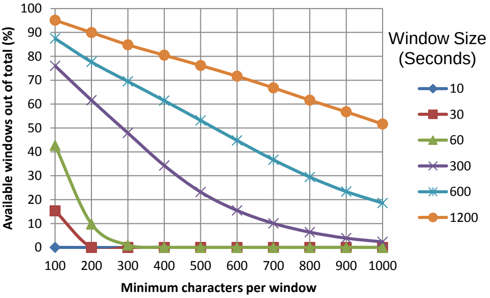
Active Authentication – Method



Active Authentication – Eval

Availability by minimum char thresholds:

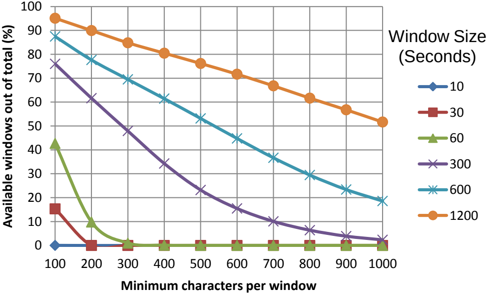
- Larger window ⇒ higher decision availability
- Windows < 5 mins – not very useful



Active Authentication – Eval

Availability by minimum char thresholds:

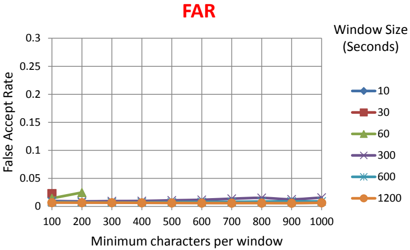
- Larger window ⇒ higher decision availability
- Windows < 5 mins – not very useful

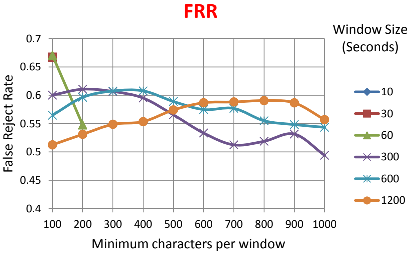


Active Authentication – Eval

Average FAR/FRR:

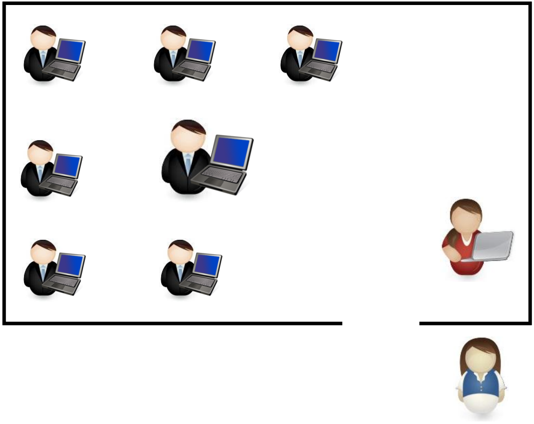
- Strict sensors
- Larger window ⇒ less affected by char/win thresholds



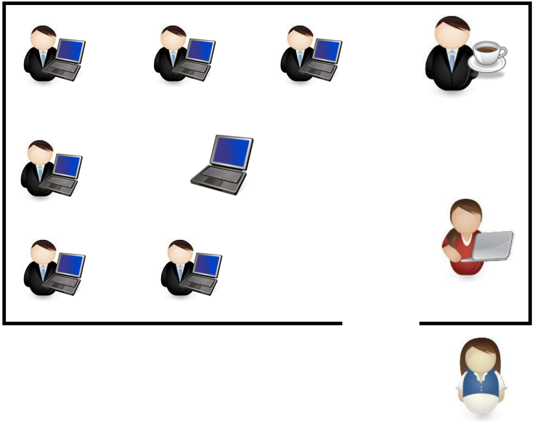


Outline

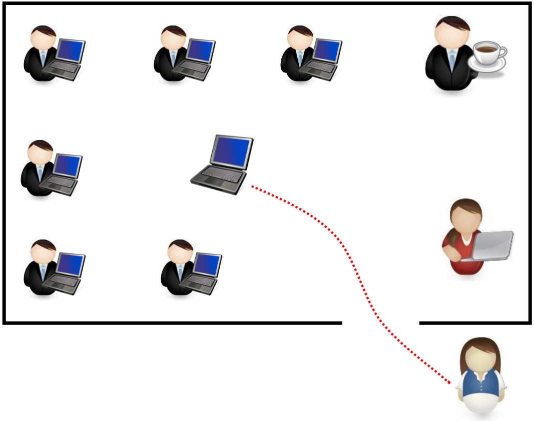
Motivation



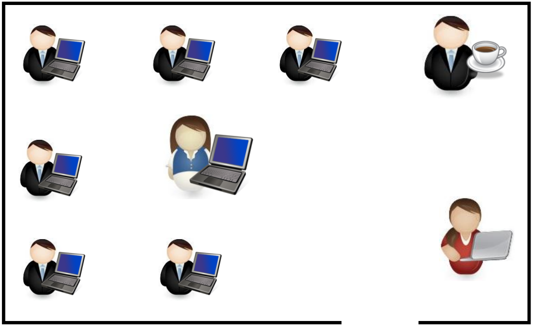
Motivation



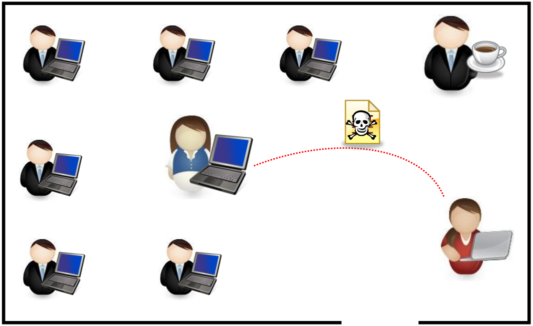
Motivation



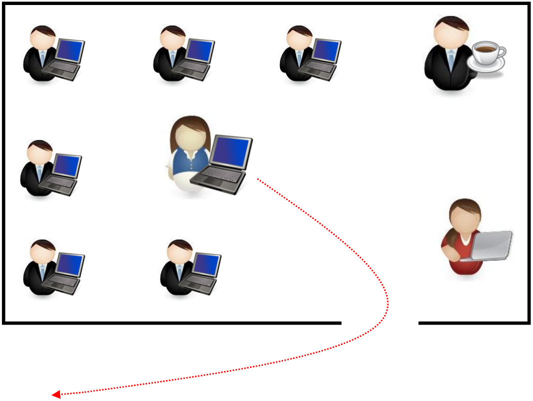
Motivation



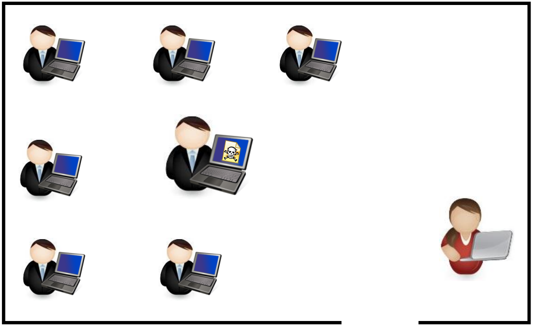
Motivation



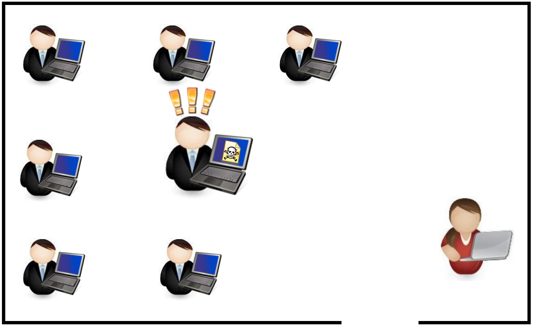
Motivation



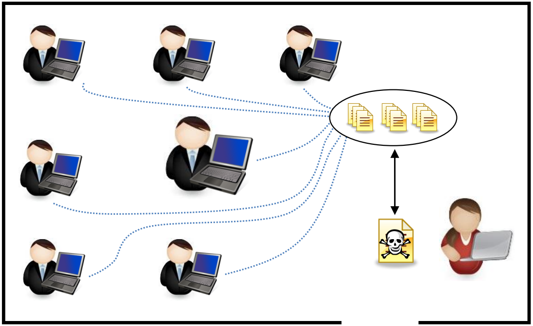
Motivation



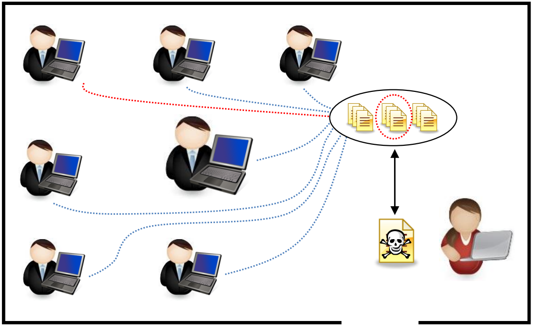
Motivation



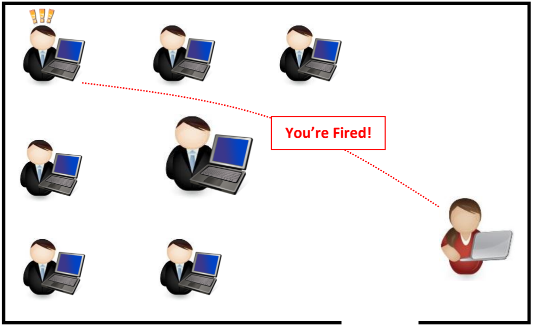
Motivation



Motivation



Motivation



Motivation – Contd.

- The web is full of anonymous communication
  - Can use stylometry to deanonymize it
- Pseudonymous documents published on the web:
  - Virtually ∞ suspects
  - Or lack of training data
- $\Rightarrow$ problem for:
  - Analysts: confidence in suspect pool
  - Users: may be falsely accused of authorship

Motivation – Contd.

- The web is full of anonymous communication
  - Can use stylometry to deanonymize it
- Pseudonymous documents published on the web:
  - Virtually ∞ suspects
  - Or lack of training data
- $\Rightarrow$ problem for:
  - Analysts: confidence in suspect pool
  - Users: may be falsely accused of authorship

Motivation – Contd.

- The web is full of anonymous communication
  - Can use stylometry to deanonymize it
- Pseudonymous documents published on the web:
  - Virtually ∞ suspects
  - Or lack of training data
- $\Rightarrow$ problem for:
  - Analysts: confidence in suspect pool
  - Users: may be falsely accused of authorship

Motivation – Contd.

- The web is full of anonymous communication
  - Can use stylometry to deanonymize it
- Pseudonymous documents published on the web:
  - Virtually ∞ suspects
  - Or lack of training data
- $\Rightarrow$ problem for:
  - Analysts: confidence in suspect pool
  - Users: may be falsely accused of authorship

Motivation – Contd.

- The web is full of anonymous communication
  - Can use stylometry to deanonymize it
- Pseudonymous documents published on the web:
  - Virtually ∞ suspects
  - Or lack of training data
- $\Rightarrow$ problem for:
  - Analysts: confidence in suspect pool
  - Users: may be falsely accused of authorship

Motivation – Contd.

- The web is full of anonymous communication
  - Can use stylometry to deanonymize it
- Pseudonymous documents published on the web:
  - Virtually ∞ suspects
  - Or lack of training data
- $\Rightarrow$ problem for:
  - Analysts: confidence in suspect pool
  - Users: may be falsely accused of authorship

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- ⇒ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Motivation – Contd.

- The *Classify-Verify* problem: mixed open/closed-world
  - Closed set of candidate authors
  - Take into account that the author may not be in the set
- $\Rightarrow$ *Classify-Verify* algorithm: classification + binary verification
  - Intercepts misclassifications
  - Tunable rigidity – FAR/FRR
  - Performs better than traditional stylometry
    - Closed-world & open-world
    - Different domains and scales
    - Adversarial settings
    - Active authentication settings

Problem Statement

- Problem building blocks – recap:
  - D: document of unknown authorship
  - $\mathcal{A} = \{A_1, ..., A_n\}$: set of candidate authors
  - $p = Pr[A_D \in \mathcal{A}]$: probability D's author is a candidate
- $\Rightarrow$ The *Classify-Verify* Problem:
  - Find D's author in $\mathcal{A}$ or determine $A_D \notin \mathcal{A}$
    - Optional: given p
- Notations:
  - *in-set* : documents whose author is a candidate ($= p$)
  - *not-in-set* : documents whose author is missing ($= 1 - p$)

Problem Statement

- Problem building blocks – recap:
  - D: document of unknown authorship
  - $\mathcal{A} = \{A_1, ..., A_n\}$: set of candidate authors
  - $p = Pr[A_D \in \mathcal{A}]$: probability D's author is a candidate
- $\Rightarrow$ The *Classify-Verify* Problem:
  - Find D's author in $\mathcal{A}$ or determine $A_D \notin \mathcal{A}$
    - Optional: given p
- Notations:
  - *in-set* : documents whose author is a candidate ($= p$)
  - *not-in-set* : documents whose author is missing ($= 1 - p$)

Problem Statement

- Problem building blocks – recap:
  - D: document of unknown authorship
  - $\mathcal{A} = \{A_1, ..., A_n\}$: set of candidate authors
  - $p = Pr[A_D \in \mathcal{A}]$: probability D's author is a candidate
- $\Rightarrow$ The *Classify-Verify* Problem:
  - Find D's author in $\mathcal{A}$ or determine $A_D \notin \mathcal{A}$
    - Optional: given p
- Notations:
  - *in-set* : documents whose author is a candidate ($= p$)
  - *not-in-set* : documents whose author is missing ($= 1 - p$)

Problem Statement

- Problem building blocks – recap:
  - D: document of unknown authorship
  - $\mathcal{A} = \{A_1, ..., A_n\}$: set of candidate authors
  - $p = Pr[A_D \in \mathcal{A}]$: probability D's author is a candidate
- $\Rightarrow$ The *Classify-Verify* Problem:
  - Find D's author in $\mathcal{A}$ or determine $A_D \notin \mathcal{A}$
    - Optional: given p
- Notations:
  - *in-set* : documents whose author is a candidate ($= p$)
  - *not-in-set* : documents whose author is missing ($= 1 - p$)

Problem Statement

- Problem building blocks – recap:
  - D: document of unknown authorship
  - $\mathcal{A} = \{A_1, ..., A_n\}$: set of candidate authors
  - $p = Pr[A_D \in \mathcal{A}]$: probability D's author is a candidate
- $\Rightarrow$ The *Classify-Verify* Problem:
  - Find D's author in $\mathcal{A}$ or determine $A_D \notin \mathcal{A}$
    - Optional: given p
- Notations:
  - *in-set* : documents whose author is a candidate $(= p)$
  - *not-in-set* : documents whose author is missing $(= 1 - p)$

Problem Statement

- Problem building blocks – recap:
  - D: document of unknown authorship
  - $\mathcal{A} = \{A_1, ..., A_n\}$: set of candidate authors
  - $p = Pr[A_D \in \mathcal{A}]$: probability D's author is a candidate
- $\Rightarrow$ The *Classify-Verify* Problem:
  - Find D's author in $\mathcal{A}$ or determine $A_D \notin \mathcal{A}$
    - Optional: given p
- Notations:
  - *in-set* : documents whose author is a candidate $(= p)$
  - *not-in-set* : documents whose author is missing $(= 1 - p)$

Problem Statement

- Problem building blocks – recap:
  - D: document of unknown authorship
  - $\mathcal{A} = \{A_1, ..., A_n\}$: set of candidate authors
  - $p = Pr[A_D \in \mathcal{A}]$: probability D's author is a candidate
- $\Rightarrow$ The *Classify-Verify* Problem:
  - Find D's author in $\mathcal{A}$ or determine $A_D \notin \mathcal{A}$
    - Optional: given p
- Notations:
  - *in-set* : documents whose author is a candidate ($= p$)
  - *not-in-set* : documents whose author is missing ($= 1 - p$)

Problems in Closed-World Models

- Closed-world models applied in open-world settings: Classifier always outputs an author
  - Chosen author is merely least-worst choice
  - Absence of true author from pool is unknown
- Extremely relevant for stylometry in online domains

Problems in Closed-World Models

- Closed-world models applied in open-world settings: Classifier always outputs an author
  - Chosen author is merely least-worst choice
  - Absence of true author from pool is unknown
- Extremely relevant for stylometry in online domains

Problems in Closed-World Models

- Closed-world models applied in open-world settings: Classifier always outputs an author
  - Chosen author is merely least-worst choice
  - Absence of true author from pool is unknown
- Extremely relevant for stylometry in online domains

Problems in Closed-World Models

- Closed-world models applied in open-world settings: Classifier always outputs an author
  - Chosen author is merely least-worst choice
  - Absence of true author from pool is unknown
- Extremely relevant for stylometry in online domains

Corpora

- Brennan-Greenstadt Adversarial Corpus (*EBG*) [BAG12]
  - 45 authors, > 6500 words each
  - Adversarial documents: deliberate style change
- ICWSM 2009 Spinn3r Blog dataset [BJS09]
  - 44M blogs, previously used for web-scale stylometry
  - Using 2 subsets, > 7500 words per author
    - $BLOG_S$: 50 authors, used as control to avoid overfitting on *EBG*
    - $BLOG_L$: 911 authors, used for large-scale evaluation
- Active Linguistic Authentication Dataset (*AAUTH*) [JNJS+13]
  - 67 users, continuous keyboard input stream

Corpora

- Brennan-Greenstadt Adversarial Corpus (*EBG*) [BAG12]
  - 45 authors, > 6500 words each
  - Adversarial documents: deliberate style change
- ICWSM 2009 Spinn3r Blog dataset [BJS09]
  - 44M blogs, previously used for web-scale stylometry
  - Using 2 subsets, > 7500 words per author
    - $BLOG_S$: 50 authors, used as control to avoid overfitting on *EBG*
    - $BLOG_L$: 911 authors, used for large-scale evaluation
- Active Linguistic Authentication Dataset (*AAUTH*) [JNJS+13]
  - 67 users, continuous keyboard input stream

Corpora

- Brennan-Greenstadt Adversarial Corpus (*EBG*) [BAG12]
  - 45 authors, > 6500 words each
  - Adversarial documents: deliberate style change
- ICWSM 2009 Spinn3r Blog dataset [BJS09]
  - 44M blogs, previously used for web-scale stylometry
  - Using 2 subsets, > 7500 words per author
    - $BLOG_S$: 50 authors, used as control to avoid overfitting on *EBG*
    - $BLOG_L$: 911 authors, used for large-scale evaluation
- Active Linguistic Authentication Dataset (*AAUTH*) [JNJS+13]
  - 67 users, continuous keyboard input stream

Corpora

- Brennan-Greenstadt Adversarial Corpus (*EBG*) [BAG12]
  - 45 authors, > 6500 words each
  - Adversarial documents: deliberate style change
- ICWSM 2009 Spinn3r Blog dataset [BJS09]
  - 44M blogs, previously used for web-scale stylometry
  - Using 2 subsets, > 7500 words per author
    - $BLOG_S$: 50 authors, used as control to avoid overfitting on *EBG*
    - $BLOG_L$: 911 authors, used for large-scale evaluation
- Active Linguistic Authentication Dataset (*AAUTH*) [JNJS+13]
  - 67 users, continuous keyboard input stream

Corpora

- Brennan-Greenstadt Adversarial Corpus (*EBG*) [BAG12]
  - 45 authors, > 6500 words each
  - Adversarial documents: deliberate style change
- ICWSM 2009 Spinn3r Blog dataset [BJS09]
  - 44M blogs, previously used for web-scale stylometry
  - Using 2 subsets, > 7500 words per author
    - $BLOG_S$: 50 authors, used as control to avoid overfitting on *EBG*
    - $BLOG_L$: 911 authors, used for large-scale evaluation
- Active Linguistic Authentication Dataset (*AAUTH*) [JNJS+13]
  - 67 users, continuous keyboard input stream

Corpora

- Brennan-Greenstadt Adversarial Corpus (*EBG*) [BAG12]
  - 45 authors, > 6500 words each
  - Adversarial documents: deliberate style change
- ICWSM 2009 Spinn3r Blog dataset [BJS09]
  - 44M blogs, previously used for web-scale stylometry
  - Using 2 subsets, > 7500 words per author
    - $BLOG_S$: 50 authors, used as control to avoid overfitting on *EBG*
    - $BLOG_L$: 911 authors, used for large-scale evaluation
- Active Linguistic Authentication Dataset (*AAUTH*) [JNJS+13]
  - 67 users, continuous keyboard input stream

Corpora

- Brennan-Greenstadt Adversarial Corpus (*EBG*) [BAG12]
  - 45 authors, > 6500 words each
  - Adversarial documents: deliberate style change
- ICWSM 2009 Spinn3r Blog dataset [BJS09]
  - 44M blogs, previously used for web-scale stylometry
  - Using 2 subsets, > 7500 words per author
    - $BLOG_S$: 50 authors, used as control to avoid overfitting on *EBG*
    - $BLOG_L$: 911 authors, used for large-scale evaluation
- Active Linguistic Authentication Dataset (*AAUTH*) [JNJS+13]
  - 67 users, continuous keyboard input stream

Feature Set

- Tested several feature sets
  - *Writeprints* – extensive feature set
    Lexical, syntactic, content, grammar, idiosyncrasies...
  - $k \in \{50, ..., 1000\}$ most common $n \in \{1, ..., 5\}$-grams
    $\langle k, n \rangle$*-chars*, $\langle k, n \rangle$*-words*
  - $\langle 500, 2 \rangle$*-chars* wins
    Best F1-score on *EBG* & $BLOG_S$

Feature Set

- Tested several feature sets
  - *Writeprints* – extensive feature set
    Lexical, syntactic, content, grammar, idiosyncrasies...
  - $k \in \{50, ..., 1000\}$ most common $n \in \{1, ..., 5\}$-grams
    $\langle k, n \rangle$*-chars*, $\langle k, n \rangle$*-words*
  - $\langle 500, 2 \rangle$*-chars* wins
    Best F1-score on *EBG* & $BLOG_S$

Feature Set

- Tested several feature sets
  - *Writeprints* – extensive feature set
    Lexical, syntactic, content, grammar, idiosyncrasies...
  - $k \in \{50, ..., 1000\}$ most common $n \in \{1, ..., 5\}$-grams
    $\langle k, n \rangle$*-chars*, $\langle k, n \rangle$*-words*
  - $\langle 500, 2 \rangle$*-chars* wins
    Best F1-score on *EBG* & $BLOG_S$

Classify-Verify

- Abstaining classifier: refrain when not sure
- Closed-world classifier + verifier $\rightarrow$ open-world
- Output range: $\mathcal{A} \rightarrow \mathcal{A} \cup \{\perp\}$
  - $\perp =$ "unknown"
- Manual/automatically set verification threshold t
- Aim to maximize F1-scores for some expected *in-set* $\% = p$
  - p *in-set* documents
  - $1 - p$ *not-in-set* documents

Classify-Verify

- Abstaining classifier: refrain when not sure
- Closed-world classifier + verifier $\rightarrow$ open-world
- Output range: $\mathcal{A} \rightarrow \mathcal{A} \cup \{\perp\}$
  - $\perp$ = "unknown"
- Manual/automatically set verification threshold t
- Aim to maximize F1-scores for some expected *in-set* $\% = p$
  - p *in-set* documents
  - $1 - p$ *not-in-set* documents

Classify-Verify

- Abstaining classifier: refrain when not sure
- Closed-world classifier + verifier $\rightarrow$ open-world
- Output range: $\mathcal{A} \rightarrow \mathcal{A} \cup \{\perp\}$
  - $\perp =$ "unknown"
- Manual/automatically set verification threshold t
- Aim to maximize F1-scores for some expected *in-set* $\% = p$
  - p *in-set* documents
  - $1 - p$ *not-in-set* documents

Classify-Verify

- Abstaining classifier: refrain when not sure
- Closed-world classifier + verifier $\rightarrow$ open-world
- Output range: $\mathcal{A} \rightarrow \mathcal{A} \cup \{\perp\}$
  - $\perp =$ "unknown"
- Manual/automatically set verification threshold t
- Aim to maximize F1-scores for some expected *in-set* $\% = p$
  - p *in-set* documents
  - $1 - p$ *not-in-set* documents

Classify-Verify

- Abstaining classifier: refrain when not sure
- Closed-world classifier + verifier $\rightarrow$ open-world
- Output range: $\mathcal{A} \rightarrow \mathcal{A} \cup \{\perp\}$
  - $\perp =$ "unknown"
- Manual/automatically set verification threshold t
- Aim to maximize F1-scores for some expected *in-set* $\% = p$
  - p *in-set* documents
  - $1 - p$ *not-in-set* documents

Classify: Closed-World Setup

- Authorship Attribution: which $A \in \mathcal{A}$ wrote D?
- SMO SVM as underlying classifier for the "Classify" phase
- Also used to establish "classify-only" baseline
  - How closed-world classifiers perform in open-world? (not good...)

Classify: Closed-World Setup

- Authorship Attribution: which $A \in \mathcal{A}$ wrote D?
- SMO SVM as underlying classifier for the “Classify” phase
- Also used to establish “classify-only” baseline
  - How closed-world classifiers perform in open-world? (not good...)

Classify: Closed-World Setup

- Authorship Attribution: which $A \in \mathcal{A}$ wrote D?
- SMO SVM as underlying classifier for the “Classify” phase
- Also used to establish “classify-only” baseline
  - How closed-world classifiers perform in open-world? (not good...)

Verify: Open-World Setup

- Authorship Verification: is D written by A?
  - Naïve #1: reduce to 1-vs-all modeling *not-A*
  - Naïve #2: cross validate A vs D & test distinguishability
- Verification methods:
  - Classifier-induced: based on closed-world classifier outputs
    P_1, P_1-P_2-*Diff*, *Gap-Conf*
  - Standalone: models built using A's training data *only*
    V, V_σ, V_σ^a
- Also used to establish "verify-only" baseline

Verify: Open-World Setup

- Authorship Verification: is D written by A?
  - Naïve #1: reduce to 1-vs-all modeling *not*-A
  - Naïve #2: cross validate A vs D & test distinguishability
- Verification methods:
  - Classifier-induced: based on closed-world classifier outputs
    P_1, P_1-P_2-*Diff*, *Gap-Conf*
  - Standalone: models built using A's training data *only*
    V, V_σ, V_σ^a
- Also used to establish "verify-only" baseline

Verify: Open-World Setup

- Authorship Verification: is D written by A?
  - Naïve #1: reduce to 1-vs-all modeling *not-A*
  - Naïve #2: cross validate A vs D & test distinguishability
- Verification methods:
  - Classifier-induced: based on closed-world classifier outputs
    P_1, P_1-P_2-*Diff*, *Gap-Conf*
  - Standalone: models built using A's training data *only*
    V, V_σ, V_σ^a
- Also used to establish "verify-only" baseline

Verify: Open-World Setup

- Authorship Verification: is D written by A?
  - Naïve #1: reduce to 1-vs-all modeling *not*-A
  - Naïve #2: cross validate A vs D & test distinguishability
- Verification methods:
  - Classifier-induced: based on closed-world classifier outputs
    P_1, P_1-P_2-*Diff*, *Gap-Conf*
  - Standalone: models built using A's training data *only*
    V, V_σ, V_σ^a
- Also used to establish "verify-only" baseline

Verify: Open-World Setup

- Authorship Verification: is D written by A?
  - Naïve #1: reduce to 1-vs-all modeling *not-A*
  - Naïve #2: cross validate A vs D & test distinguishability
- Verification methods:
  - Classifier-induced: based on closed-world classifier outputs
    P_1, P_1-P_2-*Diff*, *Gap-Conf*
  - Standalone: models built using A's training data *only*
    V, V_σ, V_σ^a
- Also used to establish "verify-only" baseline

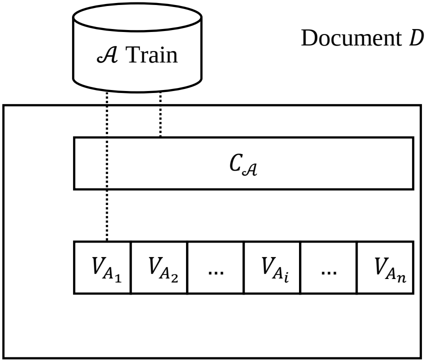
Verify: Open-World Setup

- Authorship Verification: is D written by A?
  - Naïve #1: reduce to 1-vs-all modeling *not*-A
  - Naïve #2: cross validate A vs D & test distinguishability
- Verification methods:
  - Classifier-induced: based on closed-world classifier outputs P_1, P_1-P_2-*Diff*, *Gap-Conf*
  - Standalone: models built using A's training data *only* V, V_σ, V_σ^a
- Also used to establish "verify-only" baseline

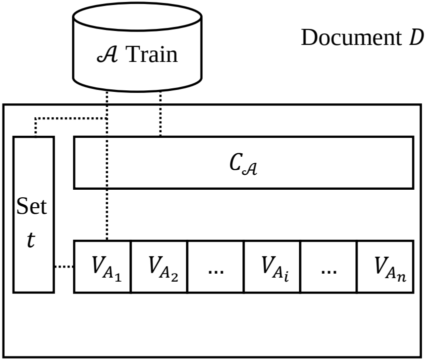
Verify: Open-World Setup

- Authorship Verification: is D written by A?
  - Naiv̈e #1: reduce to 1-vs-all modeling *not*-A
  - Naiv̈e #2: cross validate A vs D & test distinguishability
- Verification methods:
  - Classifier-induced: based on closed-world classifier outputs
    P_1, P_1-P_2-*Diff*, *Gap-Conf*
  - Standalone: models built using A's training data *only*
    V, V_σ, V_σ^a
- Also used to establish "verify-only" baseline

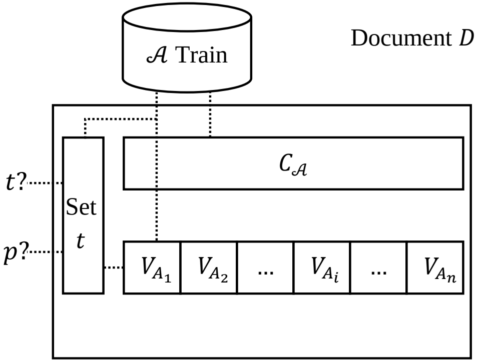
Classify-Verify – Flow



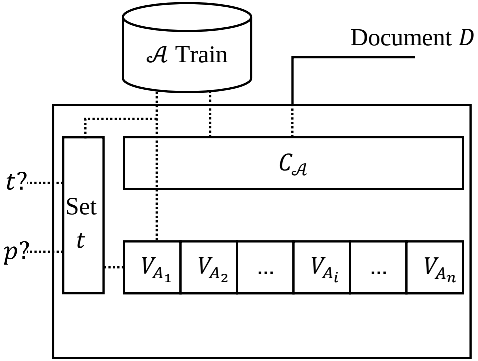
Classify-Verify – Flow



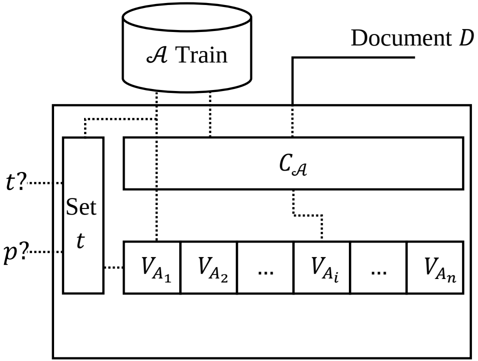
Classify-Verify – Flow



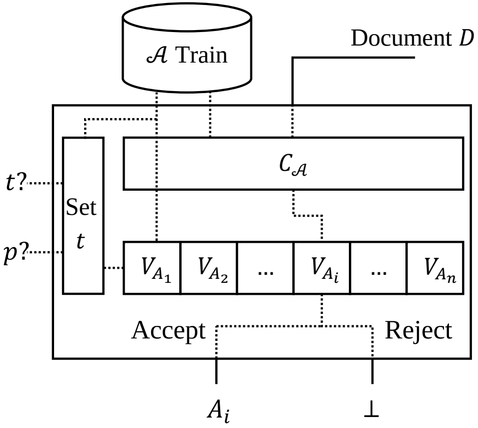
Classify-Verify – Flow



Classify-Verify – Flow



Classify-Verify – Flow



Classify-Verify – Threshold Selection

- Oracle: manually-set for best performance on test data
- p-Induced: t set empirically over training set
  - to maximize F1-scores for p
- Robust: t set empirically over training set
  - to maximize expected F1-scores over all $p \in 0.1, 0.2, ..., 1.0$

Classify-Verify – Threshold Selection

- Oracle: manually-set for best performance on test data
- *p*-Induced: *t* set empirically over training set
  - to maximize F1-scores for *p*
- Robust: *t* set empirically over training set
  - to maximize expected F1-scores over all $p \in 0.1, 0.2, ..., 1.0$

Classify-Verify – Threshold Selection

- Oracle: manually-set for best performance on test data
- *p*-Induced: *t* set empirically over training set
  - to maximize F1-scores for *p*
- Robust: *t* set empirically over training set
  - to maximize expected F1-scores over all $p \in 0.1, 0.2, ..., 1.0$

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp$ =attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp$ =attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp =$attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp =$attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp$ =attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp =$attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp$ =attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- n-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp$ =attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying p: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp$ =attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

Evaluation Methodology

- *n*-fold cross-validation
  - *EBG* adversarial: classify attack docs ($\perp =$attack)
- Baselines
  - Only closed-world classifiers
  - Only binary (standalone) verifiers
- Varying *p*: proportion/probability of *in-set* documents
  - 10%, 20%, ... $\rightarrow$ 100% (pure closed-world)
  - 10 experiments, in each only $p \times n$ authors are trained on
- *Flexible* vs. *Strict* Evaluation:
  - *Flexible* : count *all* thwarted misclassifications as *true*
  - *Strict* : count only *not-in-set* thwarted misclassification as *true*
- Measure F1-score: precision $\leftrightarrow$ recall

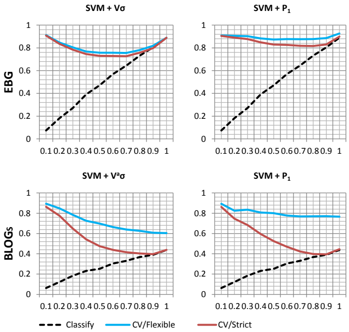
Results

Results

Results: *EBG*/*BLOGs*

Classify-Verify outperforms closed-world classifiers alone

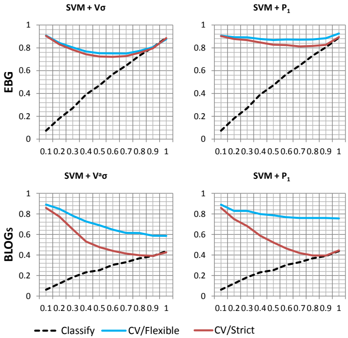
- Using oracle thresholds



Results: *EBG*/*BLOG*$_S$– *p*-Induced Thresholds

Classify-Verify outperforms closed-world classifiers alone

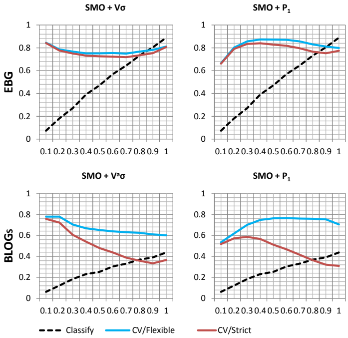
- Using *p*-induced thresholds as well – similar to oracle



Results: *EBG*/*BLOG$_S$*– *Robust* Thresholds

Classify-Verify outperforms closed-world classifiers alone

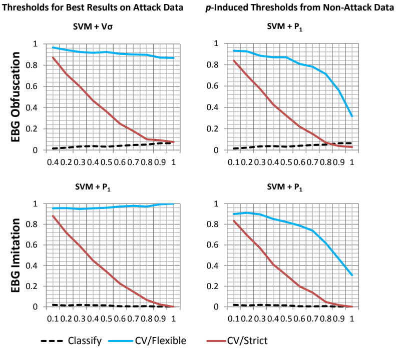
- Using *Robust* thresholds for most *in-set* scenarios, without knowing *p*!



Results: *EBG* Adversarial Settings

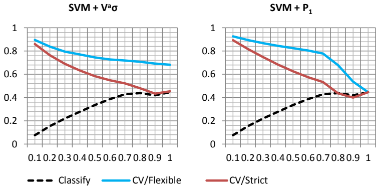
Classify-Verify successfully thwarts most attacks

- Even if thresholds not set to hold-off attacks



Results: $BLOG_L$

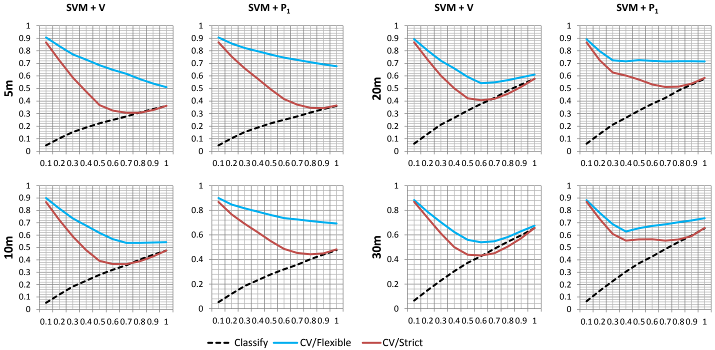
Classify-Verify outperforms closed-world models on large-scale datasets



Results: *AAUTH*

Classify-Verify outperforms closed-world models in active authentication settings

- For 5, 10, 20, 30-minute windows with 1-minute decision frequency



Classify-Verify – Conclusion

- *Classify-Verify* is effective in open-world settings
  - Also more effective in closed-world settings
  - Automatic threshold selection performs well w/ or w/o knowing p
- Effective in thwarting attacks
  - Even without special “defensive” configuration
- Effective in large-scale, open-world domain datasets
- Effective in dynamic, noisy active authentication settings
- $\Rightarrow$ *Classify-Verify* is preferable over closed-world classifiers almost always
  - Essential tool for analysis of open-world and closed-world problems

Classify-Verify – Conclusion

- *Classify-Verify* is effective in open-world settings
  - Also more effective in closed-world settings
  - Automatic threshold selection performs well w/ or w/o knowing p
- Effective in thwarting attacks
  - Even without special “defensive” configuration
- Effective in large-scale, open-world domain datasets
- Effective in dynamic, noisy active authentication settings
- $\Rightarrow$ *Classify-Verify* is preferable over closed-world classifiers almost always
  - Essential tool for analysis of open-world and closed-world problems

Classify-Verify – Conclusion

- *Classify-Verify* is effective in open-world settings
  - Also more effective in closed-world settings
  - Automatic threshold selection performs well w/ or w/o knowing p
- Effective in thwarting attacks
  - Even without special “defensive” configuration
- Effective in large-scale, open-world domain datasets
- Effective in dynamic, noisy active authentication settings
- $\Rightarrow$ *Classify-Verify* is preferable over closed-world classifiers almost always
  - Essential tool for analysis of open-world and closed-world problems

Classify-Verify – Conclusion

- *Classify-Verify* is effective in open-world settings
  - Also more effective in closed-world settings
  - Automatic threshold selection performs well w/ or w/o knowing p
- Effective in thwarting attacks
  - Even without special "defensive" configuration
- Effective in large-scale, open-world domain datasets
- Effective in dynamic, noisy active authentication settings
- $\Rightarrow$ *Classify-Verify* is preferable over closed-world classifiers almost always
  - Essential tool for analysis of open-world and closed-world problems

Classify-Verify – Conclusion

- *Classify-Verify* is effective in open-world settings
  - Also more effective in closed-world settings
  - Automatic threshold selection performs well w/ or w/o knowing p
- Effective in thwarting attacks
  - Even without special “defensive” configuration
- Effective in large-scale, open-world domain datasets
- Effective in dynamic, noisy active authentication settings
- $\Rightarrow$ *Classify-Verify* is preferable over closed-world classifiers almost always
  - Essential tool for analysis of open-world and closed-world problems

Classify-Verify – Conclusion

- *Classify-Verify* is effective in open-world settings
  - Also more effective in closed-world settings
  - Automatic threshold selection performs well w/ or w/o knowing p
- Effective in thwarting attacks
  - Even without special “defensive” configuration
- Effective in large-scale, open-world domain datasets
- Effective in dynamic, noisy active authentication settings
- ⇒ *Classify-Verify* is preferable over closed-world classifiers almost always
  - Essential tool for analysis of open-world and closed-world problems

Classify-Verify – Conclusion

- *Classify-Verify* is effective in open-world settings
  - Also more effective in closed-world settings
  - Automatic threshold selection performs well w/ or w/o knowing p
- Effective in thwarting attacks
  - Even without special "defensive" configuration
- Effective in large-scale, open-world domain datasets
- Effective in dynamic, noisy active authentication settings
- $\Rightarrow$ *Classify-Verify* is preferable over closed-world classifiers almost always
  - Essential tool for analysis of open-world and closed-world problems

Outline

Summary

- Increase in online discourse, pools of authors, countermeasures against stylometry
  - Necessitates robust, open-world stylometric methods
- Authorship verification – useful approach for security & open-world applications
  - Problem relaxation $\rightarrow$ improve classification (LFID)
  - High-level security applications (Active Authentication)
  - Open-world problems (the *Classify-Verify* algorithm)
- Verification-infused classification – shown effective in improving closed-world classifiers alone

Summary

- Increase in online discourse, pools of authors, countermeasures against stylometry
  - Necessitates robust, open-world stylometric methods
- Authorship verification – useful approach for security & open-world applications
  - Problem relaxation $\rightarrow$ improve classification (LFID)
  - High-level security applications (Active Authentication)
  - Open-world problems (the *Classify-Verify* algorithm)
- Verification-infused classification – shown effective in improving closed-world classifiers alone

Summary

- Increase in online discourse, pools of authors, countermeasures against stylometry
  - Necessitates robust, open-world stylometric methods
- Authorship verification – useful approach for security & open-world applications
  - Problem relaxation → improve classification (LFID)
  - High-level security applications (Active Authentication)
  - Open-world problems (the *Classify-Verify* algorithm)
- Verification-infused classification – shown effective in improving closed-world classifiers alone

Summary

- Increase in online discourse, pools of authors, countermeasures against stylometry
  - Necessitates robust, open-world stylometric methods
- Authorship verification – useful approach for security & open-world applications
  - Problem relaxation → improve classification (LFID)
  - High-level security applications (Active Authentication)
  - Open-world problems (the *Classify-Verify* algorithm)
- Verification-infused classification – shown effective in improving closed-world classifiers alone

Summary

- Increase in online discourse, pools of authors, countermeasures against stylometry
  - Necessitates robust, open-world stylometric methods
- Authorship verification – useful approach for security & open-world applications
  - Problem relaxation $\rightarrow$ improve classification (LFID)
  - High-level security applications (Active Authentication)
  - Open-world problems (the *Classify-Verify* algorithm)
- Verification-infused classification – shown effective in improving closed-world classifiers alone

Summary

- Increase in online discourse, pools of authors, countermeasures against stylometry
  - Necessitates robust, open-world stylometric methods
- Authorship verification – useful approach for security & open-world applications
  - Problem relaxation $\rightarrow$ improve classification (LFID)
  - High-level security applications (Active Authentication)
  - Open-world problems (the *Classify-Verify* algorithm)
- Verification-infused classification – shown effective in improving closed-world classifiers alone

Directions for Future Authorship Verification Research

- Expand and elevate authorship verification research as a preferred approach for stylometry
  - Integrating binary verification with closed-world classification
  - Expanding empirical foundations of verification evaluation
  - Fusion of verification methods
  - Verification used for security and privacy

Directions for Future Authorship Verification Research

- Expand and elevate authorship verification research as a preferred approach for stylometry
  - Integrating binary verification with closed-world classification
  - Expanding empirical foundations of verification evaluation
  - Fusion of verification methods
  - Verification used for security and privacy

Directions for Future Authorship Verification Research

- Expand and elevate authorship verification research as a preferred approach for stylometry
  - Integrating binary verification with closed-world classification
  - Expanding empirical foundations of verification evaluation
  - Fusion of verification methods
  - Verification used for security and privacy

Directions for Future Authorship Verification Research

- Expand and elevate authorship verification research as a preferred approach for stylometry
  - Integrating binary verification with closed-world classification
  - Expanding empirical foundations of verification evaluation
  - Fusion of verification methods
  - Verification used for security and privacy

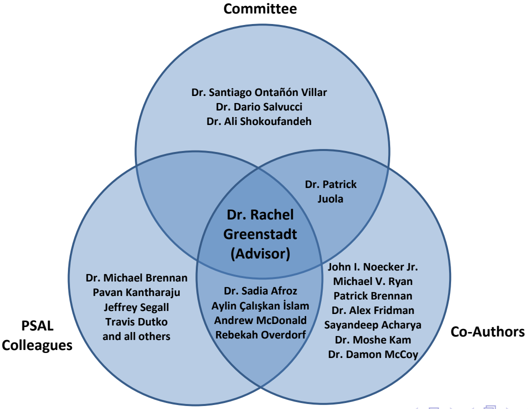
Directions for Future Authorship Verification Research

- Expand and elevate authorship verification research as a preferred approach for stylometry
  - Integrating binary verification with closed-world classification
  - Expanding empirical foundations of verification evaluation
  - Fusion of verification methods
  - Verification used for security and privacy

Thank You

Thank You!

Contact: stolerman@cs.drexel.edu
The Privacy Security & Automation Lab @ Drexel: http://psal.cs.drexel.edu/



For Further Reading I

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

Michael Brennan, Sadia Afroz, and Rachel Greenstadt.
Adversarial stylometry: Circumventing authorship recognition to preserve privacy and anonymity.
ACM Trans. Inf. Syst. Secur., 15(3):12:1–12:22, November 2012.

Kevin Burton, Akshay Java, and Ian Soboroff.
The icwsm 2009 spinn3r dataset.
In *Proceedings of the Third Annual Conference on Weblogs and Social Media (ICWSM 2009), San Jose, CA*, 2009.

Alex Fridman, Ariel Stolerman, Sayandeep Acharya, Patrick Brennan, Patrick Juola, Rachel Greenstadt, and Moshe Kam.
Decision fusion for multimodal active authentication.
IT Professional, 15(4):29–33, 2013.

Lex Fridman, Ariel Stolerman, Sayandeep Acharya, Patrick Brennan, Patrick Juola, Rachel Greenstadt, and Moshe Kam.
Multi-modal decision fusion for continuous authentication.
Computers & Electrical Engineering, (0):–, 2014.

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

For Further Reading II

Patrick Juola, John Noecker Jr., Ariel Stolerman, Michael V. Ryan, Patrick Brennan, and Rachel Greenstadt.
A dataset for active linguistic authentication.
In *Proceedings of the Ninth Annual IFIP WG 11.9 International Conference on Digital Forensics*, Orlando, Florida, USA, January 2013. National Center for Forensic Science.

Patrick Juola, John I. Noecker, Ariel Stolerman, Michael V. Ryan, Patrick Brennan, and Rachel Greenstadt.
Keyboard-behavior-based authentication.
IT Professional, 15(4):8–11, 2013.

P. Juola.
Jgaap, a java-based, modular, program for textual analysis, text categorization, and authorship attribution.

Andrew McDonald, Sadia Afroz, Aylin Caliskan, Ariel Stolerman, and Rachel Greenstadt.
Use fewer instances of the letter "i": Toward writing style anonymization.
In *Privacy Enhancing Technologies Symposium (PETS)*, 2012.

John Noecker Jr. and Michael Ryan.
Distractorless authorship verification.
In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA).

Ariel Stolerman, Aylin Caliskan, and Rachel Greenstadt.
From language to family and back: Native language and language family identification from english text.
In *Proceedings of the 2013 NAACL HLT Student Research Workshop*, pages 32–39, Atlanta, Georgia, June 2013. Association for Computational Linguistics.

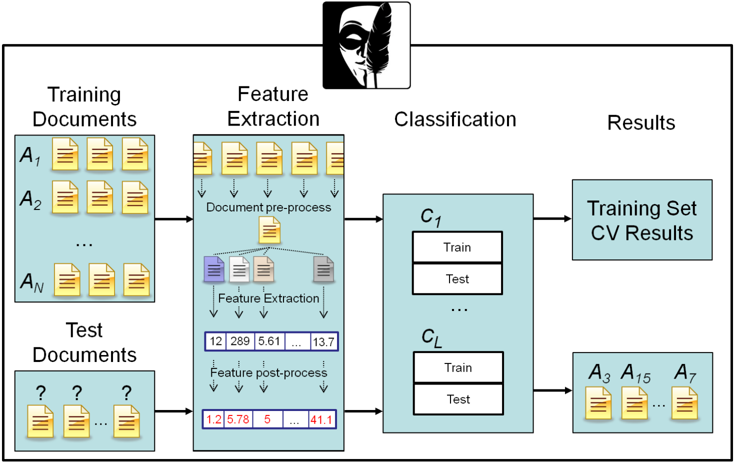
For Further Reading III

Ariel Stolerman, Alex Fridman, Rachel Greenstadt, Patrick Brennan, and Patrick Juola.
Active linguistic authentication revisited: Real-time stylometric evaluation towards multi-modal decision fusion.
In *The Tenth Annual IFIP WG 11.9 International Conference on Digital Forensics*, January 2014.

Ariel Stolerman and Rachel Greenstadt.
Mixed closed-world and open-world authorship attribution.
IEEE Transactions on Information Forensics and Security [under submission].

Ariel Stolerman, Rebekah Overdorf, Sadia Afroz, and Rachel Greenstadt.
Classify, but verify: Breaking the closed-world assumption in stylometric authorship attribution.
In *The Tenth Annual IFIP WG 11.9 International Conference on Digital Forensics*, January 2014.

JStylo: Authorship Attribution Framework



Classifier-Induced Verification

- Confidence in given solution by distance-based classifiers
- Classify $\rightarrow$ set threshold $\rightarrow$ test
- Consider $P_1 \geq P_2 \geq \ldots \geq P_n$ for $A_i \in \mathcal{A}$:
  - *P_1* : classifier's probability for chosen author
  - *P_1-P_2-Diff* : diff b/w probabilities of top and 2^{nd}-to-top authors
  - *Gap-Conf* : like P_1-P_2-*Diff*, using n 1-vs-all classifiers

Classifier-Induced Verification

- Confidence in given solution by distance-based classifiers
- Classify $\rightarrow$ set threshold $\rightarrow$ test
- Consider $P_1 \geq P_2 \geq ... \geq P_n$ for $A_i \in \mathcal{A}$:
  - P_1 : classifier's probability for chosen author
  - P_1-P_2-*Diff* : diff b/w probabilities of top and 2^{nd}-to-top authors
  - *Gap-Conf* : like P_1-P_2-*Diff*, using n 1-vs-all classifiers

Classifier-Induced Verification

- Confidence in given solution by distance-based classifiers
- Classify $\rightarrow$ set threshold $\rightarrow$ test
- Consider $P_1 \geq P_2 \geq ... \geq P_n$ for $A_i \in \mathcal{A}$:
  - *P_1* : classifier's probability for chosen author
  - *P_1-P_2-Diff* : diff b/w probabilities of top and 2^{nd}-to-top authors
  - *Gap-Conf* : like P_1-P_2-*Diff*, using n 1-vs-all classifiers

Classifier-Induced Verification

- Confidence in given solution by distance-based classifiers
- Classify $\rightarrow$ set threshold $\rightarrow$ test
- Consider $P_1 \geq P_2 \geq ... \geq P_n$ for $A_i \in \mathcal{A}$:
  - P_1 : classifier's probability for chosen author
  - P_1*-*P_2*-Diff* : diff b/w probabilities of top and 2^{nd}-to-top authors
  - *Gap-Conf* : like P_1-P_2-*Diff*, using n 1-vs-all classifiers

Classifier-Induced Verification

- Confidence in given solution by distance-based classifiers
- Classify $\rightarrow$ set threshold $\rightarrow$ test
- Consider $P_1 \geq P_2 \geq ... \geq P_n$ for $A_i \in \mathcal{A}$:
  - *P_1* : classifier's probability for chosen author
  - *P_1-P_2-Diff* : diff b/w probabilities of top and 2^{nd}-to-top authors
  - *Gap-Conf* : like *P_1-P_2-Diff*, using n 1-vs-all classifiers

Classifier-Induced Verification

- Confidence in given solution by distance-based classifiers
- Classify $\rightarrow$ set threshold $\rightarrow$ test
- Consider $P_1 \geq P_2 \geq ... \geq P_n$ for $A_i \in \mathcal{A}$:
  - P_1 : classifier's probability for chosen author
  - P_1*-*P_2*-Diff* : diff b/w probabilities of top and 2^{nd}-to-top authors
  - *Gap-Conf* : like P_1-P_2-*Diff*, using n 1-vs-all classifiers

Standalone Verification

- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for "looser" ones
  - V_σ : per-feature SD normalization
  - V^a : account for A's avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for "looser" ones
  - V_σ : per-feature SD normalization
  - V^a : account for A's avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for “looser” ones
  - V_σ : per-feature SD normalization
  - V^a : account for A's avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for "looser" ones
  - V_σ : per-feature SD normalization
  - V^a : account for A's avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

- *V*: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char *n*-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for "looser" ones
  - V_σ : per-feature SD normalization
  - V^a : account for *A*'s avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for "looser" ones
  - V_σ : per-feature SD normalization
  - V^a : account for A's avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for "looser" ones
  - V_σ : per-feature SD normalization
  - V^a : account for A's avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

- *V*: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char *n*-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for "looser" ones
  - V_σ : per-feature SD normalization
  - V^a : account for *A*'s avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

Standalone Verification

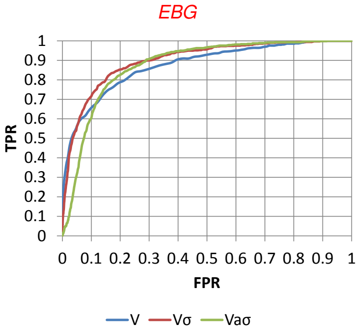
- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for “looser” ones
  - V_σ : per-feature SD normalization
  - V^a : account for A’s avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$-*chars*

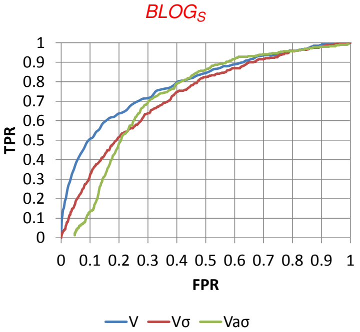
Standalone Verification

- V: Distractorless Verification [NJR12]
  - Standardize char-case & whitespaces, extract word/char n-grams
  - Author model $M = \langle m_1, m_2, ..., m_n \rangle$
  - Document model $F = \langle f_1, f_2, ..., f_n \rangle$
  - Test: $\delta(M, F) < t$?
- Variants:
  - Tighten bound for less varied authors, widen for “looser” ones
  - V_σ : per-feature SD normalization
  - V^a : account for A's avg. pairwise document distances
  - Evaluation w/ 10-fold CV + $\langle 500, 2 \rangle$*-chars*

Standalone Verification – Contd.

- ROC curves: no method is strictly preferred over the other
  - EBG (left): V_σ wins, Blog (right): V wins

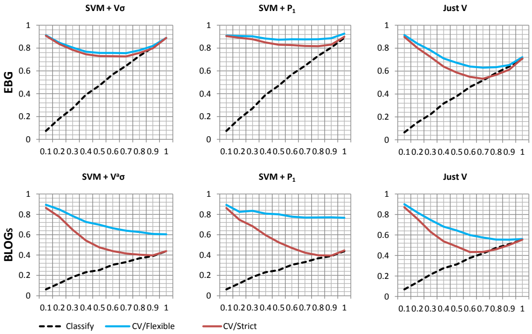




Results: *EBG*/*BLOGs*

Classify-Verify outperforms closed-world classifier *and* open-world verifiers alone

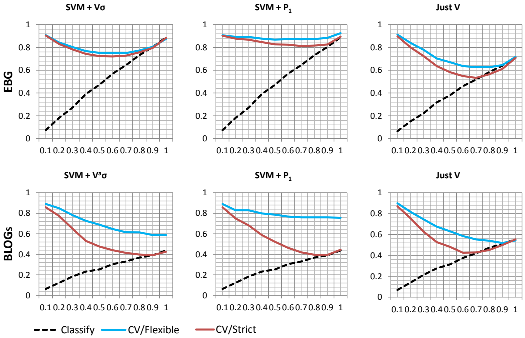
- Using oracle thresholds



Results: *EBG*/*BLOG*$_S$– *p*-Induced Thresholds

Classify-Verify outperforms closed-world classifier *and* open-world verifiers alone

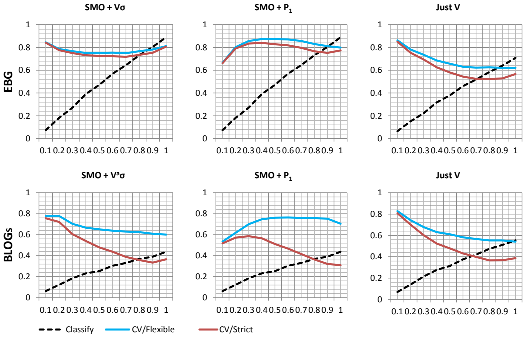
- Using *p*-induced thresholds as well – similar to oracle



Results: *EBG*/*BLOG*$_S$– *Robust* Thresholds

Classify-Verify outperforms closed-world classifier *and* open-world verifiers alone

- Using *Robust* thresholds for most *in-set* scenarios, without knowing *p*!



Results: *AAUTH*

Classify-Verify outperforms closed-world models in active authentication settings

- For 5, 10, 20, 30-minute windows with 1-minute decision frequency