Authorship Verification via k-Nearest Neighbor Estimation

Oren Halvani, Martin Steinebach, Ralf Zimmermann

Fraunhofer Institute for Secure Information Technology (SIT), Darmstadt, Germany
Department of Computer Science Technische Universität Darmstadt, Germany
OUTLINE

- Verification schemes
- Features & Feature-Categories
- Our approach
- Evaluation
- Benefits / challenges / future work
So, let’s start immediately…

To avoid repetition, please do not make introductions or motivations of the task. Rather, immediately start with your approach, and how it differs from the state of the art (i.e., your contributions).
VERIFICATION SCHEME
(CLASSICAL VERSION...)

Training Set

Alleged document

Features

Threshold

Decision

Verification Model

\[ f_1 \quad f_2 \quad f_3 \quad f_4 \quad f_5 \]
VERIFICATION SCHEME (OUR VERSION…)

Training Set

Feature-Categories

Alleged document

Verification Model

for each: $F_i$

apply majority vote

Threshold

Decision

$F_1, F_2, F_3, F_4, F_5, ...$
FEATURES

- Features are the core of any AV system!
FEATURES

- Features are the core of any AV system!

- Usually classified into so-called linguistic layers (e.g. survey of Stamatatos)
FEATURES

- Features are the core of any AV system!

- Usually classified into so-called linguistic layers (e.g. survey of Stamatatos)

  - Semantic layer
  - Syntactic layer
  - Lexical layer
  - Character layer
  - Phoneme layer

There are even more, e.g. Layout layer
FEATURES

Features are the core of any AV system!

Usually classified into so-called linguistic layers (e.g. survey of Stamatatos)

Instead of "layers" we prefer to use the term "Feature-Categories"…
FEATURE CATEGORIES

- We understand a "Feature-Category" as a concept of features, belonging to (at least) one linguistic layer…
**FEATURE CATEGORIES**

We understand a "Feature-Category" as a concept of features, belonging to (at least) one linguistic layer...

<table>
<thead>
<tr>
<th>$F_i$</th>
<th>Feature category</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_1$</td>
<td>Punctuation marks</td>
<td>-, ., ; , , ( ), [ ], ()</td>
</tr>
<tr>
<td>$F_2$</td>
<td>Letters</td>
<td>a, b, c, ..., x, y, z, A, B, C, ..., X, Y, Z</td>
</tr>
<tr>
<td>$F_3$</td>
<td>Letter n-Grams</td>
<td>en, er, th, ted, ough</td>
</tr>
<tr>
<td>$F_4$</td>
<td>Token k-prefixes</td>
<td>[removed] $\leadsto$ [re], [confirmed] $\leadsto$ [con]</td>
</tr>
<tr>
<td>$F_5$</td>
<td>Token k-suffixes</td>
<td>[extended] $\leadsto$ [ed], [available] $\leadsto$ [able]</td>
</tr>
<tr>
<td>$F_6$</td>
<td>Function words</td>
<td>and, or, the, on, in, while</td>
</tr>
<tr>
<td>$F_7$</td>
<td>Function word n-Grams</td>
<td>(which, is, or), (that, on, the, above)</td>
</tr>
<tr>
<td>$F_8$</td>
<td>Sentence k-beginning function words</td>
<td>(The...), (Since the...)</td>
</tr>
<tr>
<td>$F_9$</td>
<td>Token n-Grams</td>
<td>(such that), (it could not)</td>
</tr>
<tr>
<td>$F_{10}$</td>
<td>Token n-Gram lengths</td>
<td>(of the) $\leadsto$ (2, 3), (are known as) $\leadsto$ (3, 5, 2)</td>
</tr>
<tr>
<td>$F_{11}$</td>
<td>Token n-Gram k-prefixes</td>
<td>(has been more) $\leadsto$ (ha, be, mo)</td>
</tr>
<tr>
<td>$F_{12}$</td>
<td>Token n-Gram k-suffixes</td>
<td>(has been more) $\leadsto$ (as, en, re)</td>
</tr>
</tbody>
</table>
Note: Majority of these Feature-Categories can be parameterized…
Note: Majority of these Feature-Categories can be parameterized…

- \textit{n-Gram sizes}
- \textit{k-prefix / suffixes}
- \textit{Amount of dictionary based features}
- \textit{etc.}
FEATURE-CATEGORIES (PARAMETERS)

- **Note:** Majority of these Feature-Categories can be parameterized…
  - *n-Gram sizes*
  - *k-prefix / suffixes*
  - *Amount of dictionary based features*
  - *etc.*

- **Moreover:** Frequencies of extracted features are also kept variable
  (e.g. “*use the 120 most frequent letter-bigrams*”)
FEATURE-CATEGORIES (PARAMETERS)

**Note:** Majority of these Feature-Categories can be parameterized...

- $n$-Gram sizes
- $k$-prefix / suffixes
- Amount of dictionary based features
- etc.

**Moreover:** Frequencies of extracted features are also kept variable (e.g. "use the 120 most frequent letter-bigrams")

**Consequence:** Practically unlimited parameter space!
Feature-Categories (Parameters)

- **Note:** Majority of these Feature-Categories can be parameterized…
  - *n*-Gram sizes
  - *k*-prefix / suffixes
  - Amount of dictionary based features
  - etc.

- **Moreover:** Frequencies of extracted features are also kept variable (e.g. “use the 120 most frequent letter-bigrams“)

- **Consequence:** Practically unlimited parameter space!

- **(Unsatisfactory) solution:** random examination…
OUR APPROACH

- The procedure of our AV system can be divided into three steps:
OUR APPROACH

The procedure of our AV system can be divided into three steps:

- Preprocessing
OUR APPROACH

The procedure of our AV system can be divided into three steps:

1. Preprocessing
2. Compute style deviation scores
OUR APPROACH

- The procedure of our AV system can be divided into three steps:

1. Preprocessing
2. Compute style deviation scores
3. Determine verification decision
OUR APPROACH: PREPROCESSING

- Applying preprocessing in terms of normalization and noise reduction
OUR APPROACH: PREPROCESSING

Applying preprocessing in terms of **normalization** and **noise reduction**

- Essential to treat all documents uniquely!

  - e.g. substituting diacritics, successive blanks, etc.
OUR APPROACH: PREPROCESSING

Applying preprocessing in terms of **normalization** and **noise reduction**

- Essential to treat all documents uniquely!
  - e.g. substituting diacritics, successive blanks, etc.

- Important to increase quality of extracted features!
  - e.g. removing citations, markup-tags, formulas, non-words, etc.
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Our approach is based on a k-Nearest Neighbours (k-NN) classifier
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Our approach is based on a k-Nearest Neighbours (k-NN) classifier

- Hence, we need to construct feature-vectors from $Y$ and $X_1, X_2, ..., X_m$
  $\rightarrow$ for each chosen Feature-Category…

.
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Our approach is based on a k-Nearest Neighbours (k-NN) classifier.

- Hence, we need to construct feature-vectors from $Y$ and $X_1, X_2, ..., X_m$ for each chosen Feature-Category...
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Our approach is based on a k-Nearest Neighbours (k-NN) classifier

- Hence, we need to construct feature-vectors from \( Y \) and \( X_1, X_2, ..., X_m \)
  \( \rightarrow \) for each chosen Feature-Category…

- Alleged document
- All documents from the training set
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Our approach is based on a k-Nearest Neighbours (k-NN) classifier

- Hence, we need to construct feature-vectors from $Y$ and $X_1, X_2, \ldots, X_m$ → for each chosen Feature-Category…

- Important: Majority-voting needs an uneven number of individual decisions → hence, number of $F_i$ is always odd
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- We calculate pairwise style deviation scores (SDS) between Y and X₁, X₂, ..., Xₘ for each chosen Fᵢ.
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- We calculate pairwise style deviation scores (SDS) between $Y$ and $X_1, X_2, ..., X_m$ for each chosen $F_i$

- A SDS is a number between $[0 - \infty)$ which is calculated through a distance function, e.g. Euclidean distance:
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

We calculate pairwise style deviation scores (SDS) between $Y$ and $X_1, X_2, ..., X_m$ for each chosen $F_i$

A SDS is a number between $[0 - \infty)$ which is calculated through a distance function, e.g. Euclidean distance:

$$dist_{Euclid}(X, Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- We calculate pairwise style deviation scores (SDS) between Y and X₁, X₂, ..., Xₘ for each chosen Fᵢ.

- A SDS is a number between [0 - ∞) which is calculated through a distance function, e.g. Euclidean distance:

  \[ \text{dist}_{\text{Euclid}}(X, Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \]

- The closer a SDS is to zero, the more similar \( X_i \) is to Y.
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- We calculate pairwise style deviation scores (SDS) between $Y$ and $X_1, X_2, \ldots, X_m$ for each chosen $F_i$

- A SDS is a number between $[0 - \infty]$ which is calculated through a distance function, e.g. Euclidean distance:

$$dist_{Euclid}(X, Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- The closer a SDS is to zero, the more similar $X_i$ is to $Y$

- Once all SDS‘s are calculated we’ve got to store them…
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Resulting SDS’s are stored together with the corresponding feature vectors into a sorted list (ascending order, according to the scores)
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Resulting SDS’s are stored together with the corresponding feature vectors into a sorted list (ascending order, according to the scores)

\[
Outer_Distances = ( (SDS_1, X_1), (SDS_2, X_2), ..., (SDS_m, X_m) )
\]
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Resulting SDS’s are stored together with the corresponding feature vectors into a sorted list (ascending order, according to the scores)

\[
\text{Outer Distances} = \left\{ (SDS_1, X_1), (SDS_2, X_2), \ldots, (SDS_m, X_m) \right\}
\]

- Next, we extract the first tuple and calculate again SDS’s but now between \(X_1\) and \(X_2, X_3, \ldots, X_m\)
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Resulting SDS’s are stored together with the corresponding feature vectors into a sorted list (ascending order, according to the scores)

\[
\text{Outer\_Distances} = ( (SDS_1, X_1), (SDS_2, X_2), ..., (SDS_m, X_m) )
\]

- Next, we extract the first tuple and calculate again SDS’s but now between \(X_1\) and \(X_2, X_3, ..., X_m\)

- Now we store only the SDS’s into another ordered list:
OUR APPROACH: COMPUTE STYLE DEVIATION SCORES

- Resulting SDS’s are stored together with the corresponding feature vectors into a sorted list (ascending order, according to the scores)

\[
\text{Outer\_Distances} = ( (SDS_1, X_1), (SDS_2, X_2), \ldots, (SDS_m, X_m) )
\]

- Next, we extract the first tuple and calculate again SDS’s but now between \(X_1\) and \(X_2, X_3, \ldots, X_m\)

- Now we store only the SDS’s into another ordered list:

\[
\text{Inner\_Distances} = ( SDS_2, SDS_3, \ldots, SDS_m )
\]
OUR APPROACH: DETERMINE VERIFICATION DECISION

To obtain a decision regarding a chosen feature category we first calculate the average of the $k$-SDS's within $\text{Inner\_Distances}$:
OUR APPROACH: DETERMINE VERIFICATION DECISION

- To obtain a decision regarding a chosen feature category we first calculate the average of the \( k \)-SDS's within \( \text{Inner\_Distances} \):

\[
\text{avg\_SDS} = \frac{\text{SDS}_2 + \text{SDS}_3 + \cdots + \text{SDS}_k}{k}
\]
OUR APPROACH: DETERMINE VERIFICATION DECISION

To obtain a decision regarding a chosen feature category we first calculate the average of the $k$-SDS‘s within $Inner_Distances$:

$$avg_{SDS} = \frac{SDS_2 + SDS_3 + \cdots + SDS_k}{k}$$

$k$-NN of $X_1$
OUR APPROACH: DETERMINE VERIFICATION DECISION

- To obtain a decision regarding a chosen feature category we first calculate the average of the $k$-SDS’s within $\text{Inner\_Distances}$:

$$avg\_SDS = \frac{SDS_2 + SDS_3 + \cdots + SDS_k}{k}$$

- Now we can define an acceptance criterion.

$k$-NN of $X_1$
OUR APPROACH: DETERMINE VERIFICATION DECISION

To obtain a decision regarding a chosen feature category we first calculate the average of the $k$-SDS’s within Inner_Distances:

$$\text{avg}_{SDS} = \frac{SDS_2 + SDS_3 + \cdots + SDS_k}{k}$$

Now we can define an acceptance criterion

Accept the alleged authorship if…
OUR APPROACH: DETERMINE VERIFICATION DECISION

- To obtain a decision regarding a chosen feature category we first calculate the average of the $k$-SDS’s within $Inner_Distances$:

$$avg_{SDS} = \frac{SDS_2 + SDS_3 + \cdots + SDS_k}{k}$$

- Now we can define an acceptance criterion

- Accept the alleged authorship if…

$$\frac{SDS_1}{avg_{SDS}} \leq \text{Threshold}$$
OUR APPROACH: DETERMINE VERIFICATION DECISION

- To obtain a decision regarding a chosen feature category we first calculate the average of the $k$-SDS's within $\text{Inner\_Distances}$:

$$\text{avg\_SDS} = \frac{SDS_2 + SDS_3 + \cdots + SDS_k}{k}$$

- Now we can define an acceptance criterion

- Accept the alleged authorship if…

$$\frac{SDS_1}{\text{avg\_SDS}} \leq \text{Threshold}$$

In most of the cases: 1 performs very well…
OUR APPROACH: DETERMINE VERIFICATION DECISION

Overall decision regarding all Feature-Categories would then be:
OUR APPROACH: DETERMINE VERIFICATION DECISION

- Overall decision regarding all Feature-Categories would then be:

\[ \text{Determine verification decision} \]
OUR APPROACH: DETERMINE VERIFICATION DECISION

- Overall decision regarding all Feature-Categories would then be:

![Diagram showing decision points for F1, F2, and F3](image-url)
OUR APPROACH: DETERMINE VERIFICATION DECISION

- Overall decision regarding all Feature-Categories would then be:

\[
\begin{align*}
F_1 & \quad F_2 & \quad F_3 \\
\text{Determine verification decision} & \\
\text{Apply majority vote…} & = \\
\end{align*}
\]
EVALUATION: USED MEASURES

Simple accuracy:

\[
\varnothing = \frac{\varnothing_{C_{GR}} + \varnothing_{C_{EN}} + \cdots}{|C_{GR} \cup C_{EN} \cup \ldots|}, \text{ with } \varnothing_{C_i} = \frac{\text{Number of correct answers per dataset } C_i}{\text{Total number of documents per dataset } C_i}
\]
EVALUATION: USED MEASURES

Simple accuracy:

\[
\varnothing = \frac{\varnothing_{C_{GR}} + \varnothing_{C_{EN}} + \ldots}{|C_{GR} \cup C_{EN} \cup \ldots|}, \quad \text{with } \varnothing_{C_{i}} = \frac{\text{Number of correct answers per dataset } C_{i}}{\text{Total number of documents per dataset } C_{i}}
\]

Weighted accuracy:

\[
(\text{weighted}) \varnothing = \frac{|C_{GR}| \cdot \varnothing_{C_{GR}} + |C_{EN}| \cdot \varnothing_{C_{EN}} + \ldots}{|C_{GR} \cup C_{EN} \cup \ldots|}
\]
EVALUATION: TRAIN SET
(PAN ONLY)

- Evaluation results according to "PAN13-AI-Training Corpus"
EVALUATION: TRAIN SET (PAN ONLY)

- Evaluation results according to "PAN13-AI-Training Corpus"

<table>
<thead>
<tr>
<th>$F$</th>
<th>$\emptyset_{C_{SP}}$</th>
<th>$\emptyset_{C_{EN}}$</th>
<th>$\emptyset_{C_{GR}}$</th>
<th>$\emptyset$ (weighted) $\emptyset$</th>
</tr>
</thead>
<tbody>
<tr>
<td>{ $F_1, F_3, F_9$ }</td>
<td>80%</td>
<td>90%</td>
<td>70%</td>
<td>80%</td>
</tr>
<tr>
<td>{ $F_1, F_3, F_7, F_8, F_{12}$ }</td>
<td>80%</td>
<td>80%</td>
<td>65%</td>
<td>75%</td>
</tr>
<tr>
<td>{ $F_1, F_2, F_3$ }</td>
<td>80%</td>
<td>80%</td>
<td>55%</td>
<td>71.67%</td>
</tr>
<tr>
<td>{ $F_1, F_4, F_9$ }</td>
<td>80%</td>
<td>80%</td>
<td>60%</td>
<td>73.33%</td>
</tr>
<tr>
<td>{ $F_1, F_3, F_9, F_{11}, F_{12}$ }</td>
<td>80%</td>
<td>80%</td>
<td>55%</td>
<td>71.67%</td>
</tr>
<tr>
<td>{ $F_7, F_9, F_{11}$ }</td>
<td>60%</td>
<td>60%</td>
<td>50%</td>
<td>56.67%</td>
</tr>
<tr>
<td>{ $F_3, F_6, F_7, F_{11}, F_{12}$ }</td>
<td>60%</td>
<td>50%</td>
<td>55%</td>
<td>55%</td>
</tr>
<tr>
<td>{ $F_2, F_5, F_6$ }</td>
<td>80%</td>
<td>40%</td>
<td>40%</td>
<td>53.33%</td>
</tr>
<tr>
<td>{ $F_3, F_7, F_9$ }</td>
<td>20%</td>
<td>70%</td>
<td>50%</td>
<td>46.67%</td>
</tr>
<tr>
<td>{ $F_4, F_6, F_7$ }</td>
<td>40%</td>
<td>40%</td>
<td>60%</td>
<td>46.67%</td>
</tr>
</tbody>
</table>
EVALUATION: TRAIN SET (PAN ONLY)

- Evaluation results according to "PAN13-AI-Training Corpus"

<table>
<thead>
<tr>
<th>$\mathbf{F}$</th>
<th>$\mathcal{\phi}_{CSP}$</th>
<th>$\mathcal{\phi}_{CEN}$</th>
<th>$\mathcal{\phi}_{CGR}$</th>
<th>$\mathcal{\phi}$ (weighted) $\mathcal{\phi}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>${ F_1, F_3, F_9 }$</td>
<td>80%</td>
<td>90%</td>
<td>70%</td>
<td>80%</td>
</tr>
<tr>
<td>${ F_1, F_3, F_7, F_8, F_{12} }$</td>
<td>80%</td>
<td>80%</td>
<td>65%</td>
<td>75%</td>
</tr>
<tr>
<td>${ F_1, F_2, F_3 }$</td>
<td>80%</td>
<td>80%</td>
<td>55%</td>
<td>71.67%</td>
</tr>
<tr>
<td>${ F_1, F_4, F_9 }$</td>
<td>80%</td>
<td>80%</td>
<td>60%</td>
<td>73.33%</td>
</tr>
<tr>
<td>${ F_1, F_3, F_9, F_{11}, F_{12} }$</td>
<td>80%</td>
<td>80%</td>
<td>55%</td>
<td>71.67%</td>
</tr>
<tr>
<td>${ F_7, F_9, F_{11} }$</td>
<td>60%</td>
<td>60%</td>
<td>50%</td>
<td>56.67%</td>
</tr>
<tr>
<td>${ F_3, F_6, F_7, F_{11}, F_{12} }$</td>
<td>60%</td>
<td>50%</td>
<td>55%</td>
<td>55%</td>
</tr>
<tr>
<td>${ F_2, F_5, F_6 }$</td>
<td>80%</td>
<td>40%</td>
<td>40%</td>
<td>53.33%</td>
</tr>
<tr>
<td>${ F_3, F_7, F_9 }$</td>
<td>20%</td>
<td>70%</td>
<td>50%</td>
<td>46.67%</td>
</tr>
<tr>
<td>${ F_4, F_6, F_7 }$</td>
<td>40%</td>
<td>40%</td>
<td>60%</td>
<td>46.67%</td>
</tr>
</tbody>
</table>

- **Note:** the first one is the **best** $\mathbf{F}_i$ - combination out of $2^{12} = 4096$
EVALUATION: TRAIN SET (PAN + GERMAN CORPUS)

- Evaluation results according to "PAN13-AI-Training Corpus" in addition to a self-compiled german corpus (40 problem-cases)
EVALUATION: TRAIN SET
(PAN + GERMAN CORPUS)

Evaluation results according to "PAN13-AI-Training Corpus" in addition to a self-compiled german corpus (40 problem-cases)

<table>
<thead>
<tr>
<th>F</th>
<th>$\emptyset_{c_{SP}}$</th>
<th>$\emptyset_{c_{EN}}$</th>
<th>$\emptyset_{c_{GR}}$</th>
<th>$\emptyset_{c_{DE}}$</th>
<th>$\emptyset$ (weighted)</th>
</tr>
</thead>
<tbody>
<tr>
<td>${F_1, F_3, F_9}$</td>
<td>80 %</td>
<td>90 %</td>
<td>70 %</td>
<td>67.5 %</td>
<td>76.86 %</td>
</tr>
<tr>
<td>${F_1, F_3, F_7, F_8, F_{12}}$</td>
<td>80 %</td>
<td>80 %</td>
<td>65 %</td>
<td>77.5 %</td>
<td>75.63 %</td>
</tr>
<tr>
<td>${F_1, F_2, F_3}$</td>
<td>80 %</td>
<td>80 %</td>
<td>55 %</td>
<td>75 %</td>
<td>72.5 %</td>
</tr>
<tr>
<td>${F_1, F_4, F_9}$</td>
<td>80 %</td>
<td>80 %</td>
<td>60 %</td>
<td>62.5 %</td>
<td>70.63 %</td>
</tr>
<tr>
<td>${F_1, F_3, F_9, F_{11}, F_{12}}$</td>
<td>80 %</td>
<td>80 %</td>
<td>55 %</td>
<td>62.5 %</td>
<td>69.38 %</td>
</tr>
<tr>
<td>${F_7, F_9, F_{11}}$</td>
<td>60 %</td>
<td>60 %</td>
<td>50 %</td>
<td>60 %</td>
<td>57.5 %</td>
</tr>
<tr>
<td>${F_3, F_6, F_7, F_{11}, F_{12}}$</td>
<td>60 %</td>
<td>50 %</td>
<td>55 %</td>
<td>62.5 %</td>
<td>56.88 %</td>
</tr>
<tr>
<td>${F_2, F_5, F_6}$</td>
<td>80 %</td>
<td>40 %</td>
<td>40 %</td>
<td>65 %</td>
<td>56.26 %</td>
</tr>
<tr>
<td>${F_3, F_7, F_9}$</td>
<td>20 %</td>
<td>70 %</td>
<td>50 %</td>
<td>67.5 %</td>
<td>51.86 %</td>
</tr>
<tr>
<td>${F_4, F_6, F_7}$</td>
<td>40 %</td>
<td>40 %</td>
<td>60 %</td>
<td>60 %</td>
<td>50 %</td>
</tr>
</tbody>
</table>
EVALUATION: TRAIN SET
(PAN \rightarrow INFLUENCE OF PARAMETERS)

Evaluation results according to "PAN13-AI-Training Corpus"
with the best combination $\{F_1, F_3, F_9\}$ and various parameter-settings
EVALUATION: TRAIN SET
(PAN → INFLUENCE OF PARAMETERS)

Evaluation results according to "PAN13-AI-Training Corpus" with the best combination \( \{ F_1, F_3, F_9 \} \) and various parameter-settings

<table>
<thead>
<tr>
<th>( F_3 ), n-Gram</th>
<th>( F_3 ), Top-t</th>
<th>( F_9 ), n-Gram</th>
<th>( F_9 ), Top-t</th>
<th>( \varnothing_{c_{SP}} )</th>
<th>( \varnothing_{c_{EN}} )</th>
<th>( \varnothing_{c_{GR}} )</th>
<th>( \varnothing ) (weighted) ( \varnothing )</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>100</td>
<td>2</td>
<td>all</td>
<td>80 %</td>
<td>90 %</td>
<td>70 %</td>
<td>80 %</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
<td>2</td>
<td>all</td>
<td>80 %</td>
<td>100 %</td>
<td>65.50 %</td>
<td>82.67 %</td>
</tr>
<tr>
<td>7</td>
<td>100</td>
<td>2</td>
<td>all</td>
<td>80 %</td>
<td>80 %</td>
<td>70 %</td>
<td>76.67 %</td>
</tr>
<tr>
<td>6</td>
<td>200</td>
<td>2</td>
<td>all</td>
<td>80 %</td>
<td>100 %</td>
<td>55 %</td>
<td>78.33 %</td>
</tr>
<tr>
<td>7</td>
<td>100</td>
<td>2</td>
<td>160</td>
<td>80 %</td>
<td>80 %</td>
<td>60 %</td>
<td>73.33 %</td>
</tr>
<tr>
<td>7</td>
<td>100</td>
<td>2</td>
<td>160</td>
<td>80 %</td>
<td>80 %</td>
<td>55 %</td>
<td>71.67 %</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>2</td>
<td>all</td>
<td>80 %</td>
<td>100 %</td>
<td>40 %</td>
<td>73.33 %</td>
</tr>
<tr>
<td>3</td>
<td>all</td>
<td>2</td>
<td>all</td>
<td>60 %</td>
<td>80 %</td>
<td>55 %</td>
<td>65 %</td>
</tr>
<tr>
<td>2</td>
<td>all</td>
<td>2</td>
<td>all</td>
<td>80 %</td>
<td>80 %</td>
<td>45 %</td>
<td>68.33 %</td>
</tr>
<tr>
<td>6</td>
<td>all</td>
<td>2</td>
<td>all</td>
<td>40 %</td>
<td>80 %</td>
<td>50 %</td>
<td>56.67 %</td>
</tr>
</tbody>
</table>
EVALUATION: TEST SET

PAN 2013
Author Identification
June 12, 2013

Performances on all test data

<table>
<thead>
<tr>
<th>Submission</th>
<th>F₁</th>
<th>Precision</th>
<th>Recall</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>seidman13</td>
<td>0.753</td>
<td>0.753</td>
<td>0.753</td>
<td>65476823</td>
</tr>
<tr>
<td>halvan13</td>
<td>0.718</td>
<td>0.718</td>
<td>0.718</td>
<td>8362</td>
</tr>
<tr>
<td>layton13</td>
<td>0.671</td>
<td>0.671</td>
<td>0.671</td>
<td>9488</td>
</tr>
<tr>
<td>petmanson13</td>
<td>0.671</td>
<td>0.671</td>
<td>0.671</td>
<td>36214445</td>
</tr>
<tr>
<td>jankowska13</td>
<td>0.659</td>
<td>0.659</td>
<td>0.659</td>
<td>240335</td>
</tr>
<tr>
<td>ayala13</td>
<td>0.659</td>
<td>0.659</td>
<td>0.659</td>
<td>5577420</td>
</tr>
<tr>
<td>bobicev13</td>
<td>0.655</td>
<td>0.663</td>
<td>0.647</td>
<td>1713966</td>
</tr>
<tr>
<td>feng13</td>
<td>0.647</td>
<td>0.647</td>
<td>0.647</td>
<td>84413233</td>
</tr>
<tr>
<td>vladimir13</td>
<td>0.612</td>
<td>0.612</td>
<td>0.612</td>
<td>32608</td>
</tr>
<tr>
<td>ghaeini13</td>
<td>0.606</td>
<td>0.671</td>
<td>0.553</td>
<td>125655</td>
</tr>
<tr>
<td>vandam13</td>
<td>0.600</td>
<td>0.600</td>
<td>0.600</td>
<td>9461</td>
</tr>
<tr>
<td>moreau13</td>
<td>0.600</td>
<td>0.600</td>
<td>0.600</td>
<td>7798010</td>
</tr>
<tr>
<td>jayapal13</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>7008</td>
</tr>
<tr>
<td>grozea13</td>
<td>0.553</td>
<td>0.553</td>
<td>0.553</td>
<td>406755</td>
</tr>
<tr>
<td>gillam13</td>
<td>0.541</td>
<td>0.541</td>
<td>0.541</td>
<td>419495</td>
</tr>
<tr>
<td>kern13</td>
<td>0.529</td>
<td>0.529</td>
<td>0.529</td>
<td>624366</td>
</tr>
<tr>
<td>baseline</td>
<td>0.500</td>
<td>0.500</td>
<td>0.500</td>
<td>–</td>
</tr>
<tr>
<td>petmanson13</td>
<td>0.448</td>
<td>0.700</td>
<td>0.329</td>
<td>20671346</td>
</tr>
<tr>
<td>zhenshi13</td>
<td>0.417</td>
<td>0.800</td>
<td>0.282</td>
<td>962598</td>
</tr>
<tr>
<td>sorin13</td>
<td>0.331</td>
<td>0.633</td>
<td>0.224</td>
<td>3643942</td>
</tr>
</tbody>
</table>
## EVALUATION: TEST SET

### PAN 2013

**Author Identification**

**June 12, 2013**

**Performances on all test data**

<table>
<thead>
<tr>
<th>Submission</th>
<th>F₁</th>
<th>Precision</th>
<th>Recall</th>
<th>Runtime</th>
</tr>
</thead>
<tbody>
<tr>
<td>seidman13</td>
<td>0.753</td>
<td>0.753</td>
<td>0.753</td>
<td>65476823</td>
</tr>
<tr>
<td><strong>halvani13</strong></td>
<td><strong>0.718</strong></td>
<td><strong>0.718</strong></td>
<td><strong>0.718</strong></td>
<td><strong>8362</strong></td>
</tr>
<tr>
<td>layton13</td>
<td>0.671</td>
<td>0.671</td>
<td>0.671</td>
<td>9483</td>
</tr>
<tr>
<td>petmanson13</td>
<td>0.671</td>
<td>0.671</td>
<td>0.671</td>
<td>36214445</td>
</tr>
<tr>
<td>jankowska13</td>
<td>0.659</td>
<td>0.659</td>
<td>0.659</td>
<td>240335</td>
</tr>
<tr>
<td>ayala13</td>
<td>0.659</td>
<td>0.659</td>
<td>0.659</td>
<td>5577420</td>
</tr>
<tr>
<td>bobicev13</td>
<td>0.655</td>
<td>0.663</td>
<td>0.647</td>
<td>1713966</td>
</tr>
<tr>
<td>feng13</td>
<td>0.647</td>
<td>0.647</td>
<td>0.647</td>
<td>84413233</td>
</tr>
<tr>
<td>vladimir13</td>
<td>0.612</td>
<td>0.612</td>
<td>0.612</td>
<td>320508</td>
</tr>
<tr>
<td>ghaeini13</td>
<td>0.606</td>
<td>0.671</td>
<td>0.553</td>
<td>125655</td>
</tr>
<tr>
<td>vandam13</td>
<td>0.600</td>
<td>0.600</td>
<td>0.600</td>
<td>9461</td>
</tr>
<tr>
<td>moreau13</td>
<td>0.600</td>
<td>0.600</td>
<td>0.600</td>
<td>7798010</td>
</tr>
<tr>
<td>jayapal13</td>
<td>0.576</td>
<td>0.576</td>
<td>0.576</td>
<td>7008</td>
</tr>
<tr>
<td>grozea13</td>
<td>0.553</td>
<td>0.553</td>
<td>0.553</td>
<td>406755</td>
</tr>
<tr>
<td>gillam13</td>
<td>0.541</td>
<td>0.541</td>
<td>0.541</td>
<td>419495</td>
</tr>
<tr>
<td>kern13</td>
<td>0.529</td>
<td>0.529</td>
<td>0.529</td>
<td>624366</td>
</tr>
<tr>
<td>baseline</td>
<td>0.500</td>
<td>0.500</td>
<td>0.500</td>
<td>–</td>
</tr>
<tr>
<td>petmanson13</td>
<td>0.448</td>
<td>0.700</td>
<td>0.329</td>
<td>20671346</td>
</tr>
<tr>
<td>zhenshi13</td>
<td>0.417</td>
<td>0.800</td>
<td>0.282</td>
<td>962598</td>
</tr>
<tr>
<td>sorin13</td>
<td>0.331</td>
<td>0.633</td>
<td>0.224</td>
<td>3643942</td>
</tr>
</tbody>
</table>

If runtime would count too... 😊
BENEFITS

- Our approach has several benefits, as for instance:
Our approach has several benefits, as for instance:

Language-independent, but not cross-lingual, e.g.:
\( Y \) is written in another language than \( X_1, X_2, \ldots, X_m \).
BENEFITS

- Our approach has several benefits, as for instance:

  - **Language-independent**, but not cross-lingual, e.g.:
    
    \[ Y \text{ is written in another language than } X_1, X_2, ..., X_m \]

  - **Very fast**, there's no need for time-consuming NLP-operations
BENEFITS

- Our approach has several benefits, as for instance:

- **Language-independent**, but not cross-lingual, e.g.:
  \( Y \) is written in another language than \( X_1, X_2, \ldots, X_m \)

- **Very fast**, there's no need for time-consuming NLP-operations

- **Scalable approach**, almost anything can be replaced, expanded or combined…
Our approach has several benefits, as for instance:

- **Language-independent**, but not cross-lingual, e.g.: Y is written in another language than $X_1, X_2, \ldots, X_m$

- **Very fast**, there's no need for time-consuming NLP-operations

- **Scalable approach**, almost anything can be replaced, expanded or combined…

  Threshold, distance function(s), Feature-Categories (and their parameters),…
Biggest challenge:
Inscrutability of the methods parameter-space 😞
→ Number of parameter-settings of the feature categories is near infinite
CHALLENGES / FUTURE WORK

- Biggest challenge:
  Inscrutability of the methods parameter-space 😞
  → Number of parameter-settings of the feature categories is near infinite

- Possible solution:
  Integrate evolutionary algorithms into the AV-system to find optimal parameter settings → bad run-time performance 😞
Biggest challenge: Inscrutability of the methods parameter-space 😞 → Number of parameter-settings of the feature categories is near infinite

Possible solution: Integrate evolutionary algorithms into the AV-system to find optimal parameter settings → bad run-time performance 😞

Another challenge: Does the topic of the test (or training documents) has a strong influence on the classification result? → Still an open question...
CHALLENGES / FUTURE WORK

- Biggest challenge:
  Inscrutability of the methods parameter-space 😞
  → Number of parameter-settings of the feature categories is near infinite

- Possible solution:
  Integrate evolutionary algorithms into the AV-system to find optimal parameter settings → bad run-time performance 😞

- Another challenge:
  Does the topic of the test (or training documents) has a strong influence on the classification result? → Still an open question…

- Possible Solution:
  One of our students is currently writing his thesis to answer this question
Thank you very much for your attention!
**USED PARAMETER-SETTINGS**

What kind of parameters were used for PAN and the german corpus...?

<table>
<thead>
<tr>
<th>$F_i$</th>
<th>n-Gram</th>
<th>$k$-prefix/suffix</th>
<th>Top-$t$ (features)</th>
<th>Dictionary entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_1$</td>
<td>—</td>
<td>—</td>
<td>all</td>
<td>18 per language</td>
</tr>
<tr>
<td>$F_2$</td>
<td>—</td>
<td>—</td>
<td>all</td>
<td>$\approx 50$ per language</td>
</tr>
<tr>
<td>$F_3$</td>
<td>7</td>
<td>—</td>
<td>100</td>
<td>—</td>
</tr>
<tr>
<td>$F_4$</td>
<td>—</td>
<td>2</td>
<td>all</td>
<td>—</td>
</tr>
<tr>
<td>$F_5$</td>
<td>—</td>
<td>3</td>
<td>all</td>
<td>—</td>
</tr>
<tr>
<td>$F_6$</td>
<td>—</td>
<td>—</td>
<td>all</td>
<td>$\approx 200$ per language</td>
</tr>
<tr>
<td>$F_7$</td>
<td>—</td>
<td>—</td>
<td>all</td>
<td>—</td>
</tr>
<tr>
<td>$F_8$</td>
<td>—</td>
<td>—</td>
<td>all</td>
<td>—</td>
</tr>
<tr>
<td>$F_9$</td>
<td>2</td>
<td>—</td>
<td>all</td>
<td>—</td>
</tr>
<tr>
<td>$F_{10}$</td>
<td>3</td>
<td>2</td>
<td>160</td>
<td>—</td>
</tr>
<tr>
<td>$F_{11}$</td>
<td>3</td>
<td>2</td>
<td>200</td>
<td>—</td>
</tr>
<tr>
<td>$F_{12}$</td>
<td>3</td>
<td>3</td>
<td>200</td>
<td>—</td>
</tr>
</tbody>
</table>