ChunkedHCs Algorithm for Authorship Verification Problems: Reddit Case Study

By:
Anh Duc Le (Munster Technological University and Rigr AI), Justin McGuinness (Munster Technological University), and Edward Dixon (Rigr AI)

From the proceedings of

The Digital Forensic Research Conference

DFRWS USA 2021

July 12-15, 2021

CHUNKEDHCS ALGORITHM FOR AUTHORSHIP VERIFICATION PROBLEMS: REDDIT CASE STUDY

Anh Duc Le[a,b], Justin McGuinness[a], Edward Dixon[b]
[a] Department of Mathematics, Munster Technological University, Cork, Ireland
[b] Rigr AI, Cork, Ireland

JULY 12-15, 2021

OUR TEAM

Dr Justin McGuinness
Lecturer
Department of Mathematics
Munster Technological University
Cork, Ireland
Email: Justin.McGuinness@cit.ie

Edward Dixon
Principal (Founder)
Rigr AI
The AI company for sensitive data
Cork, Ireland
Email: dixon.edward@gmail.com
Web: https://rigr.ai/

Anh Duc Le
Machine Learning Engineer
Rigr AI
Cork, Ireland
Email: dukele35@gmail.com

1. INTRODUCTION

1. INTRODUCTION

- The **objective** is to identify whether two social media accounts belong to the same person by examining comments and posts from the two accounts. It is the **authorship verification** task of identifying whether two pieces of texts are written by the same person.
- **Authorship verification** methods are either
  - **Intrinsic**, i.e. referring to deciding whether a pair of texts are written by the same person, without having additional texts by other authors
  - **Extrinsic**, i.e. referring to using external documents from other authors to make decisions.
- The study focuses on **intrinsic authorship verification**.
- Past intrinsic authorship verification approaches:
  - K-nearest neighbour (KNN) estimation by Halvani, Steinebach & Zimmermann (2013),
  - Classification and Regression Trees (CART) by Frery, Largeron & Juganaru-Mathieu (2014),
  - Differential features based on random forest by Bartoli, Dagri, Lorenzo, Medvet & Tarlao (2015) and,
  - Deep Bayes factor scoring by Boenninghoff, Rupp, Nickel & Kolossa (2020).
- The past approaches = complex feature selections or/and sophisticated pre-processing steps

1. INTRODUCTION

- **Motivations:** seeking a simpler yet more robust approach that could offer competitive verification ability → This is the reason for introducing our new authorship verification method, **ChunkedHCs**.
- **ChunkedHCs** is based on
  1. The statistical testing Higher Criticism (Donoho & Jin, 2004) and
  2. The HC-based similarity algorithm (Kipnis, 2020a & 2020b) (Kestemont et al., 2020).
- **Novelty**:
  1. There is no need to pre-process the text inputs as ChunkedHCs considers all text elements, including characters, punctuations and special characters.
  2. ChunkedHCs are not influenced by different topics and genres, implying that the algorithm is highly versatile and applicable to various contexts.
  3. The output is a similarity probability, which is directly derived from a pair of texts without having additional information or datasets.
  4. The computation is much less intense than deep learning or machine learning approaches, since no training-data phase is involved.
- **Research questions:**
  - How are ChunkedHCs employed for the authorship verification task?
  - How do ChunkedHCs perform on Reddit users' data?
- Presentation's outline: 2) Theoretical foundations, 3) ChunkedHCs, 4) Experiments, 5) Discussion & 6) Conclusion

2. THEORETICAL FOUNDATIONS

2.1. Higher Criticism (HC)

HC Donoho & Jin (2004) is a statistical test to test test a very subtle problem whether, from a large number of independent statistical tests, all of the null hypotheses are true (global null hypothesis) or some of them are not (global alternative hypothesis)

Ho: All null hypotheses are true
Ha: Not all null hypotheses are true

Suppose there is a set of N p-values π_i, where $\pi_i \in \{\pi_1, \ldots, \pi_N\}$ is the sorted p-values from π_1 (smallest) to π_N (biggest). With $\alpha \in (0,1]$ as the tuning parameter, the HC objective function $HC(i, \pi_i)$ and the HC statistic HC^* are defined as

$$HC(i, \pi_i) = \sqrt{N} \frac{i/N - \pi_i}{\sqrt{i/N\,(1 - i/N)}} \; ; \; i = 1, \ldots, N.$$

$$HC^* = \max_{1 \le i \le \alpha N} HC(i, \pi_i)$$

When HC^* is large, the set of p-values is inconsistent with the global null hypothesis, meaning there is a presence of the non-null hypotheses (Donoho & Jin, 2008).

From the HC objective function $HC(i, \pi_i)$ and the HC statistic HC^*, the index i^* where $HC(i, \pi_i)$ is maximum and correspondingly the HC threshold t_{HC}, i.e. the p-value at $(i^*)^{th}$ index, are defined as

$$i^* \leftarrow \max_{1 \le i \le \alpha N} HC(i, \pi_i)$$

$$t_{HC} = \pi_{i^*}$$

Any p-values which are smaller than the threshold t_{HC} are responsible for the magnitude of the HC statistic HC^*. The indexes $1, \ldots, i^*$ associated with the set $\{\pi_1, \ldots, \pi_{i^*}\}$ are useful for identifying important features in classification settings (Donoho & Jin, 2009).

2. THEORETICAL FOUNDATIONS

2.2. HC-based Similarity Algorithm

- Given two documents D_1 and D_2, a word w which might occur in either D_1 and D_2 or both.
- $N(w|\ D_1)$ and $N(w|\ D_2)$ are the total number of occurrences the word w in D_1 and D_2
- Specifying a vocabulary $\mathbb{W}$
- The word-frequency tables associated with D_1 and D_2 are $\{N(w|D_1),\ w \in \mathbb{W}\}$ and $\{N(w|\ D_2),\ w \in \mathbb{W}\}$ respectively
- Total number of occurrences the word w in document D_1: $x_w = N(w|D_1)$
- Total number of occurrences the word w in both documents D_1 and D_2: $n_w = N(w|\ D_1) + N(w|\ D_2)$
- Probability of obtaining the word w in document D_1: $p_w = (|D_1| - x_w)\ /\ (|D_1| + |D_2| - n_w)$
- Kipnis (2020a) proposed a binomial word allocation model with the null hypothesis Ho that the word w symmetrically occurs in both D_1 and D_2, in other words, following the binomial distribution $Bin(n_w, p_w)$ across the two documents equally.

$$Ho: N(w|D_1) \sim Bin(n_w, p_w)$$
$$Ha: N(w|D1) \not\sim Bin(n_w, p_w)$$

- The p-value π derived from x_w, n_w and p_w for the word w:

$$\pi(w|D_1, D_2) = Prob(|Bin(n_w, p_w) - n_w p_w| \geq |N(w|D_1) - n_w p_w|)$$

- Set of p-values $\{\pi(w|D_1, D_2)\}_{w \in \mathbb{W}}$ can be obtained by performing multiple tests for all words in $\mathbb{W}$.
- Sorting $\pi_i \in Sort(\{\pi(w|D_1, D_2)\}_{w \in \mathbb{W}})$ to calculate HC statistics HC^* to detect whether any words w in the vocabulary $\mathbb{W}$ does not symmetrically occur in both D_1 and D_2. Kipnis (2020a) uses HC^* as the measure of distance $d(D_1, D_2)$ between D_1 and D_2. The smaller $d(D_1, D_2)$ is, the more likely D_1 and D_2 were written by the same author; and vice versa.
- The set of distinguishing words Δ between D_1 and D_2 will be

$$HC^*, i^* \leftarrow \max_{1 \leq i \leq \alpha N} HC(i, \pi_i) \quad \Rightarrow \quad \Delta \leftarrow \{w \in \mathbb{W}: \pi(w|D_1, D_2') \leq \pi_{i^*}\};\ \Delta \subseteq \mathbb{W}$$

2. THEORETICAL FOUNDATIONS

2.2. HC-based Similarity Algorithm

Algorithm 1. HC-based Similarity

Input: Two word-frequency tables $\{N(w|D_1), w \in \mathbb{W}\}$ and $\{N(w|D_2), w \in \mathbb{W}\}$

Step 1: Performing multiple binomial tests

$|D_1| \leftarrow \sum_{w\in\mathbb{W}} N(w|D_1)$

$|D_2| \leftarrow \sum_{w\in\mathbb{W}} N(w|D_2)$

For each $w \in \mathbb{W}$:

$x_w \leftarrow N(w|D_1)$

$n_w \leftarrow N(w|D_1) + N(w|D_2)$

$p_w \leftarrow (|D_1| - x_w)/(|D_1| + |D_2| - n_w)$

$\pi(w|D_1, D_2) \leftarrow$ exact binomial test (x_w, n_w, p_w)

Step 2: Calculating Higher Criticism Statistic

$N \leftarrow |\mathbb{W}|$

$\pi_i \in \{\pi_1, \ldots, \pi_N\} \leftarrow Sort(\{\pi(w|D_1, D_2)\}_{w\in\mathbb{W}})$

$HC(i, \pi_i) \leftarrow \sqrt{N}(i/N - \pi_i)\big(i/N(1 - i/N)\big)^{-1/2}$

$HC^*, i^* \leftarrow \max_{1\le i\le \alpha N} HC(i, \pi_i)$

$d(D_1, D_2) \leftarrow HC^*$

$\Delta \leftarrow \{w \in \mathbb{W}: \pi(w|D_1, D_2) \le \pi_{i^*}\}$

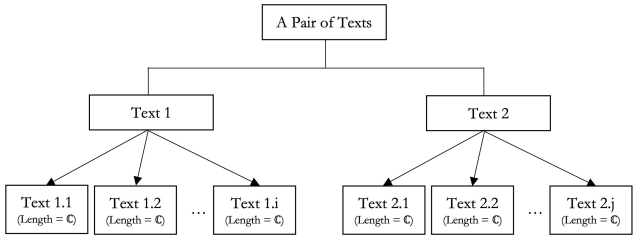
Output: $d(D_1, D_2), \Delta$

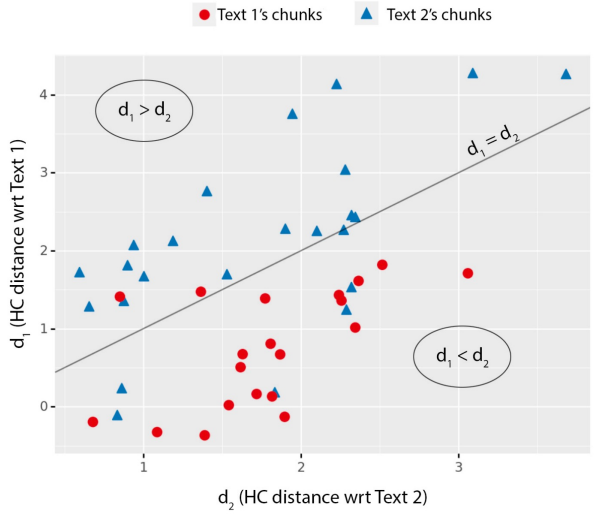
3. CHUNKEDHCS ALGORITHM

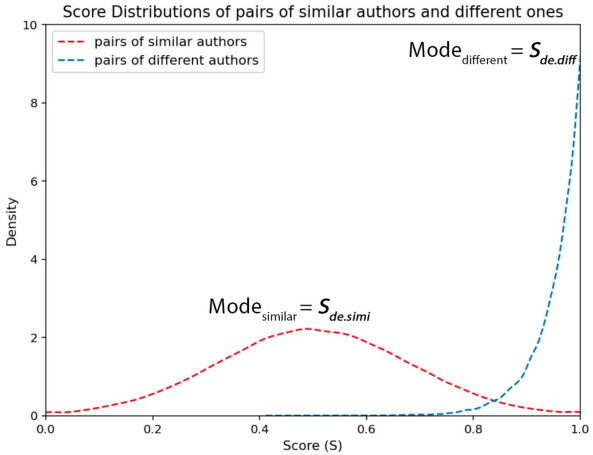
Step 1.
Chunking texts

Step 2.
Measuring HC distance between chunks and corpora

Step 3. Estimating similarity probability







3. CHUNKEDHCS ALGORITHM

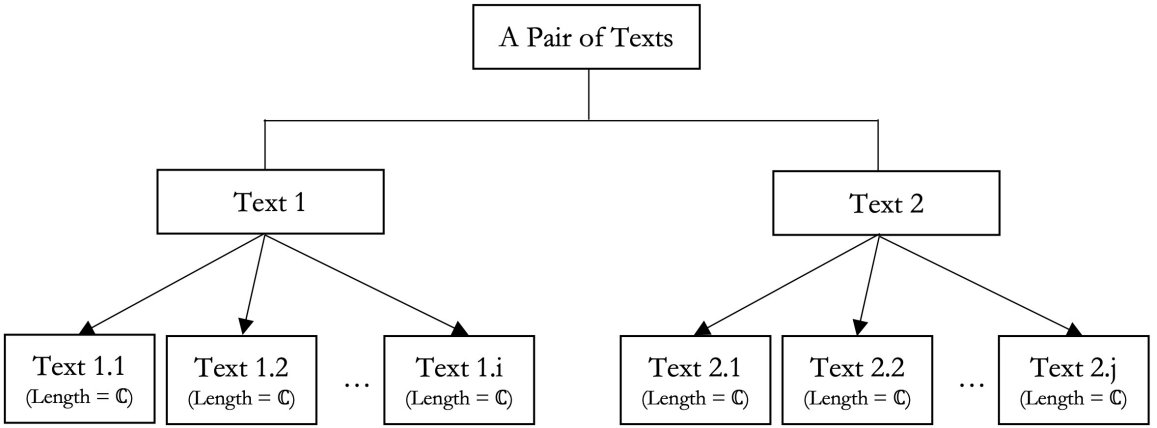
3.1. CHUNKING TEXTS (STEP 1)

- Given a pair of texts, Text 1 and Text 2, each of them is split into chunks of identical lengths of characters.
- Text 1 is split into a set of chunks $C_1 = \{T_{11}, T_{12}, \ldots, T_{1i}\}$, where C_1 is the Text 1's corpus.
- Similarly, Text 2 is also turned into a set of chunked texts $C_2 = \{T_{21}, T_{22}, \ldots, T_{2j}\}$, where C_2 is the corpus of Text 2
- The number of the chunked texts for corpus C_1 and C_2 are i and j respectively, i.e. $|C_1| = i$ and $|C_2| = j$.
- The sizes of all chunked texts are $\mathbb{C}$

$$\text{Text 1} \xrightarrow{split} C_1 = \{T_{11}, T_{12}, \ldots, T_{1i}\}; |C_1| = i$$

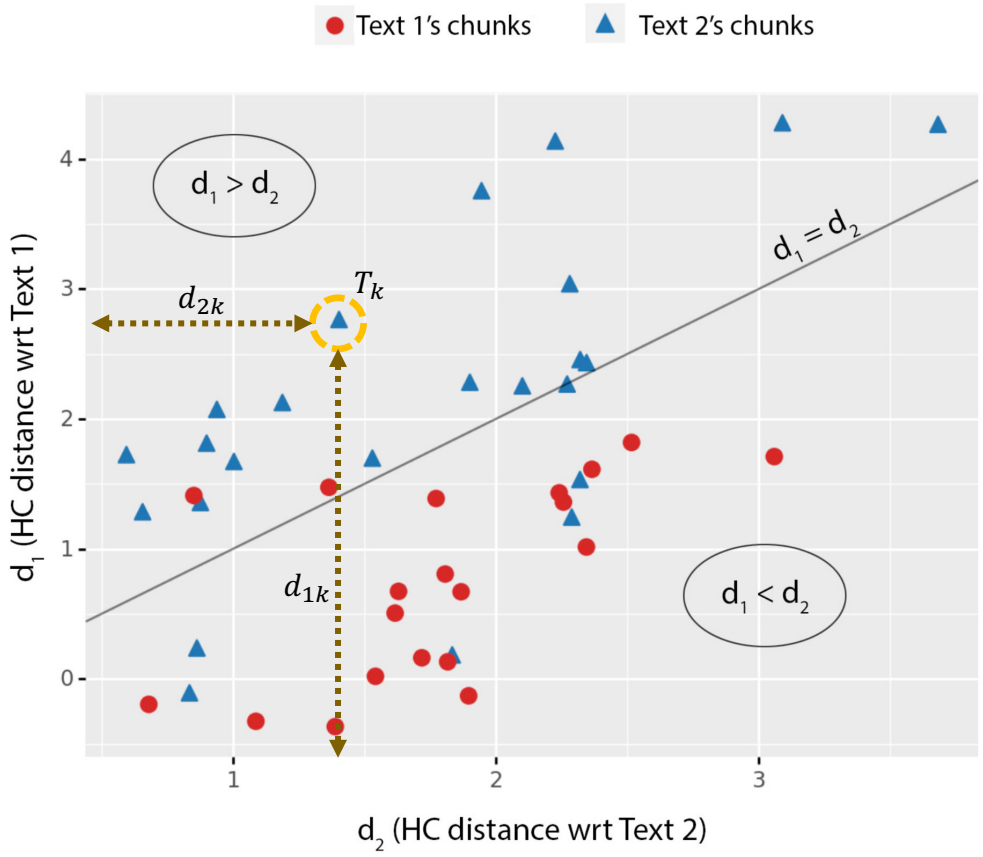
$$\text{Text 2} \xrightarrow{split} C_2 = \{T_{21}, T_{22}, \ldots, T_{2j}\}; |C_2| = j$$

$$|T| = \mathbb{C}$$



3. CHUNKEDHCS ALGORITHM

3.2. HC DISTANCE BET. CHUNKS & CORPORA (STEP 2)



$$T_k \in (C_1 \cup C_2) \Longrightarrow d_{1k} = \begin{cases} d_{HC}(T_k, C_1); & T_k \notin C_1 \\ d_{HC}(T_k, C_1^*); & T_k \in C_1 \end{cases} \quad C_1^* = C_1 \setminus \{T_k\};\ T_k \in C_1$$

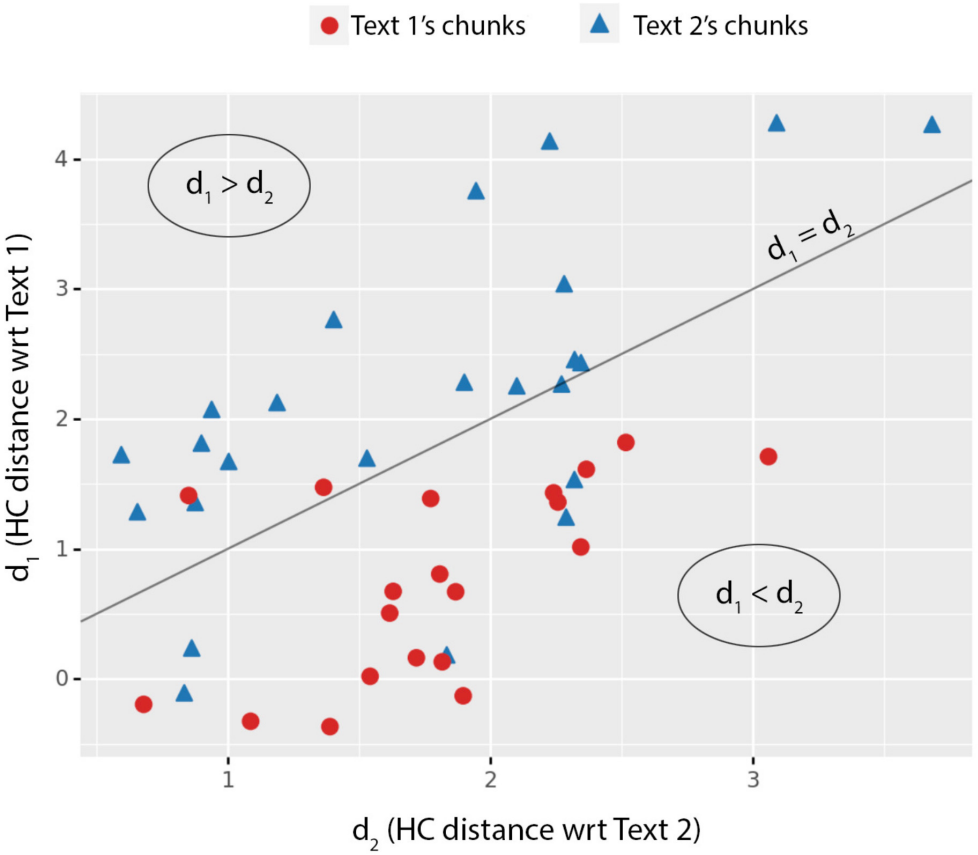
$$\& \quad d_{2k} = \begin{cases} d_{HC}(T_k, C_2); & T_k \notin C_2 \\ d_{HC}(T_k, C_2^*); & T_k \in C_2 \end{cases} \quad C_2^* = C_2 \setminus \{T_k\};\ T_k \in C_2$$

$$\Longrightarrow \begin{cases} d_{1k} < d_{2k} \Rightarrow C_{1pred} \leftarrow T_k, \\ d_{1k} > d_{2k} \Rightarrow C_{2pred} \leftarrow T_k. \end{cases}$$

$$|C_{1pred}| + |C_{2pred}| = |C_1| + |C_2|$$

3. CHUNKEDHCS ALGORITHM

3.3. ESTIMATING SIMILARITY PROBABILITY (STEP 3)



$$|C_{1pred}| + |C_{2pred}| = |C_1| + |C_2|$$

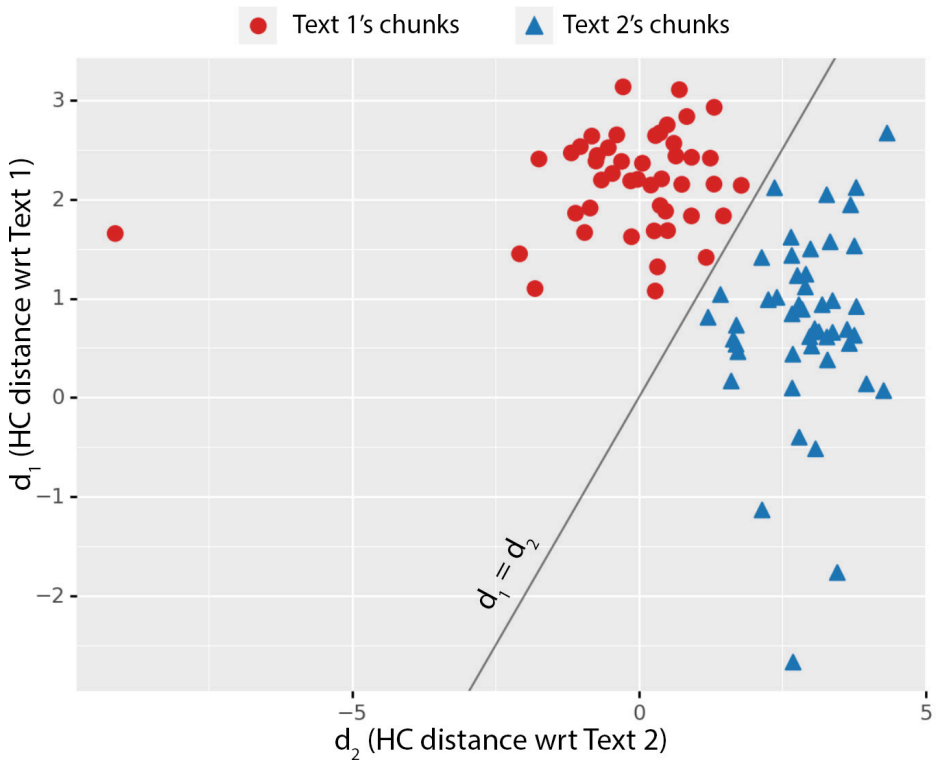
	Actual Text 1's chunks	**Actual Text 2's chunks**	Total
Predicted Text 1's chunks	True Text 1's chunks	False Text 1's chunks	$\|C_{1pred}\|$
Predicted Text 2's chunks	False Text 2's chunks	True Text 2's chunks	$\|C_{2pred}\|$
Total	$\|C_1\|$	$\|C_2\|$	

$$S = \frac{Accuracy + F1}{2};\ 0 \leq S \leq 1$$

3. CHUNKEDHCS ALGORITHM

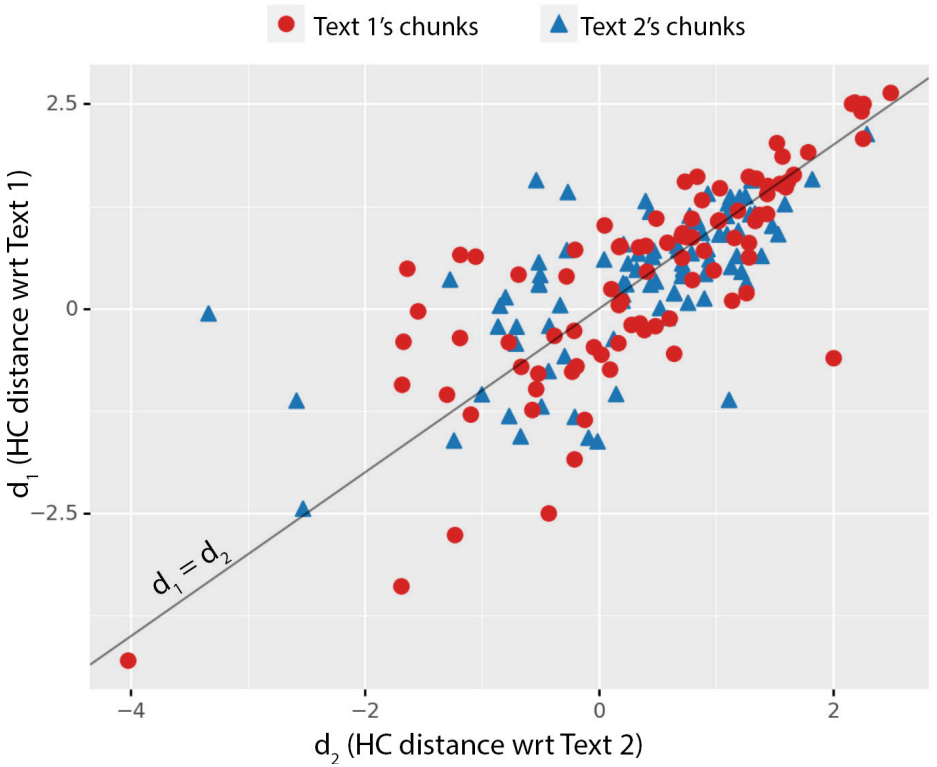
3.3. ESTIMATING SIMILARITY PROBABILITY (STEP 3)

Text 1 & Text 2 are sampled from **different authors**



$$S_{de.diff.} = \frac{Accuracy\ +\ F1}{2} = \frac{1+1}{2} = 1$$

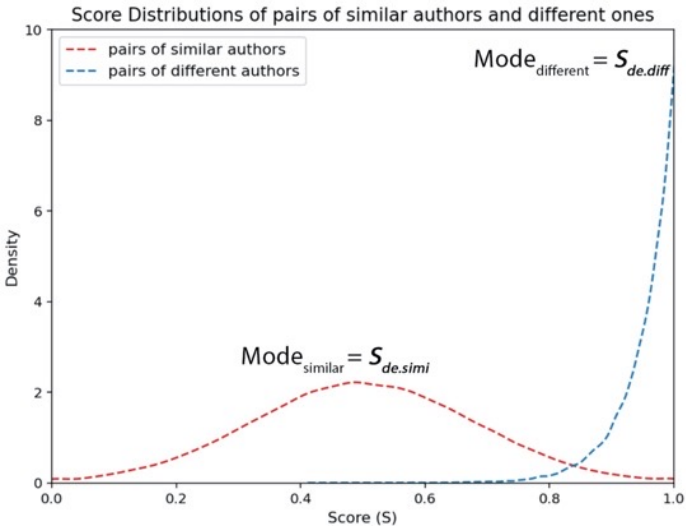
Text 1 & Text 2 are sampled from **the same author**

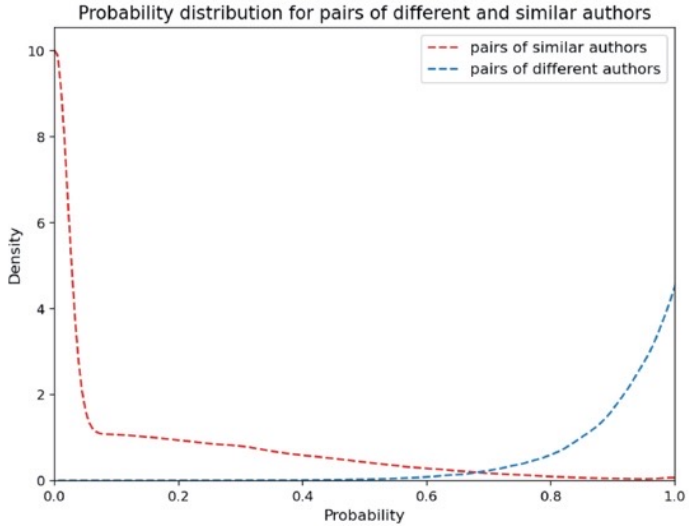


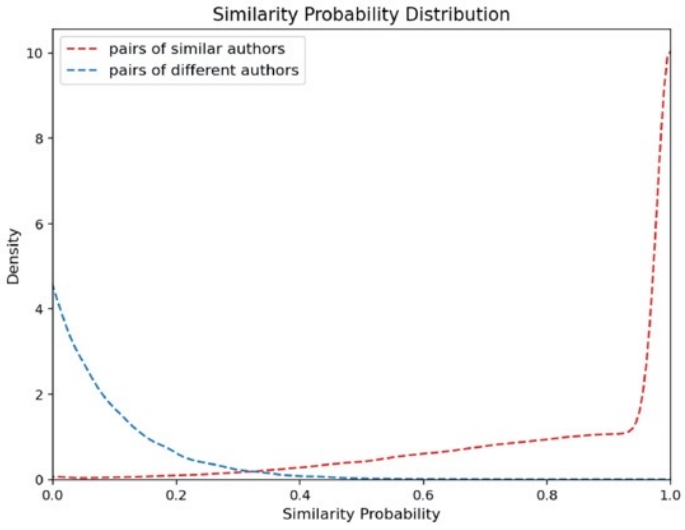
$$S_{de.simi.} = \frac{Accuracy\ +\ F1}{2} = \frac{0.5+0.5}{2} = 0.5$$

3. CHUNKEDHCS ALGORITHM

3.3. ESTIMATING SIMILARITY PROBABILITY (STEP 3)







$$S = \frac{Accuracy\ +\ F1}{2}$$

$$S_{de.diff.} = 0.5\times(1+1) = 1$$

$$S_{de.simi.} = 0.5\times(0.5+0.5) = 0.5$$

$$\Rightarrow\quad P = \begin{cases} 0 & ;\ S\ < 0.5 \\ \dfrac{S\ - S_{de.simi.}}{S_{de.diff.} - S_{de.simi.}} & ;\ S\ \geq 0.5 \end{cases} \quad\Rightarrow\quad P_{simi} = 1 - P$$

$$P_{simi} = \begin{cases} 1 & ;\ S\ < 0.5, \\ 2(1-S); & S\ \geq 0.5 \end{cases}$$

3. CHUNKEDHCS ALGORITHM

Algorithm 2. ChunkedHCs

Input: Text 1 & Text 2

Step 1: Chunk the texts

Choosing chunk size $\mathbb{C}$

$$\text{Text 1} \xrightarrow{split} C_1 = \{T_{11}, T_{12}, \ldots, T_{1i}\};\ |C_1| = i;\ |T| = \mathbb{C}$$

$$\text{Text 2} \xrightarrow{split} C_2 = \{T_{21}, T_{22}, \ldots, T_{2j}\};\ |C_2| = j;\ |T| = \mathbb{C}$$

Step 2: Measure HC distances between chunks and corpora

Choosing vocabulary size $\mathbb{V}$ for calculating d_{HC}

$$T_k \in (C_1 \cup C_2) \longrightarrow T_k \begin{cases} d_{k1} = \begin{cases} d_{HC}(T_k, C_1);\ T_k \notin C_1 \\ d_{HC}(T_k, C_1^*);\ T_k \in C_1 \end{cases} \\ d_{k2} = \begin{cases} d_{HC}(T_k, C_2);\ T_k \notin C_2 \\ d_{HC}(T_k, C_2^*);\ T_k \in C_2 \end{cases} \end{cases};$$

$$\longrightarrow \begin{cases} d_{k1} < d_{k2} \Rightarrow C_{1pred} \leftarrow T_k \\ d_{k1} > d_{k2} \Rightarrow C_{2pred} \leftarrow T_k \end{cases}$$

where $C_1^* = C_1 \setminus \{T_k\};\ T_k \in C_1$ & $C_2^* = C_2 \setminus \{T_k\};\ T_k \in C_2$

Step 3: Estimate Similarity Probability

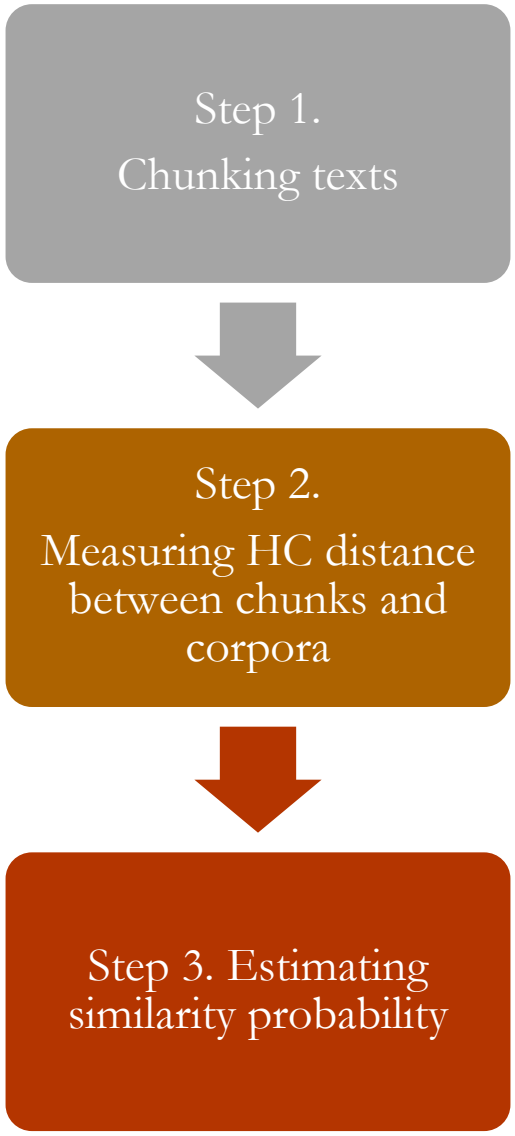
From C_1, C_2, C_{1pred} & C_{2pred}:

$$S = \frac{Accuracy\ +\ F1}{2};\ \Rightarrow P_{simi} = \begin{cases} 1 & ;\ S < 0.5 \\ 2(1-S); & S \geq 0.5 \end{cases}$$

Output:

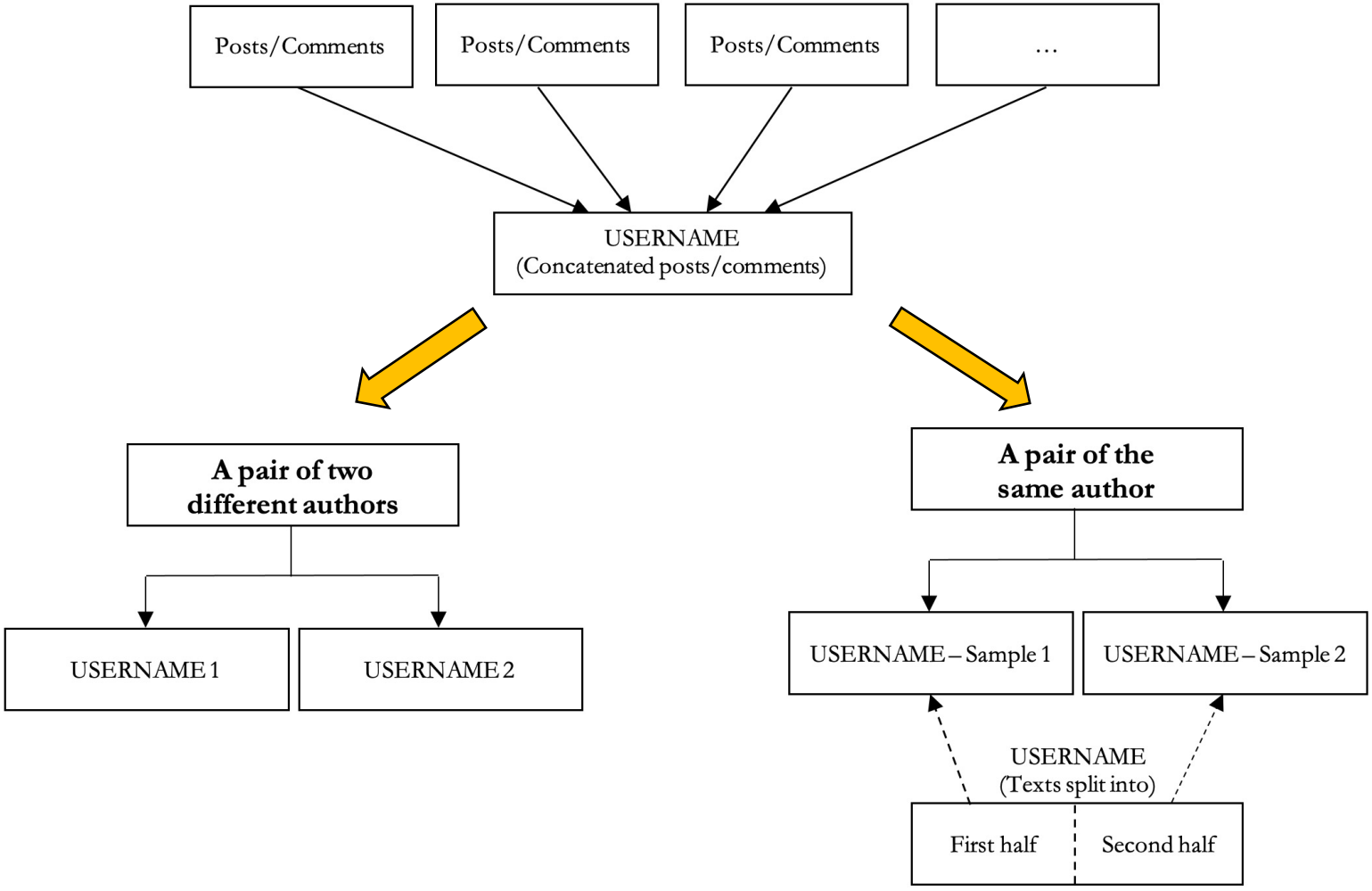
$P_{simi} \longrightarrow 1 \Rightarrow$ Text 1 and Text 2 are from the same author.

$P_{simi} \longrightarrow 0 \Rightarrow$ Text 1 and Text 2 are from different authors.



4. EXPERIMENTS

4.1. CREATING DATASETS FROM REDDIT



4. EXPERIMENTS

4.1. CREATING DATASETS FROM REDDIT

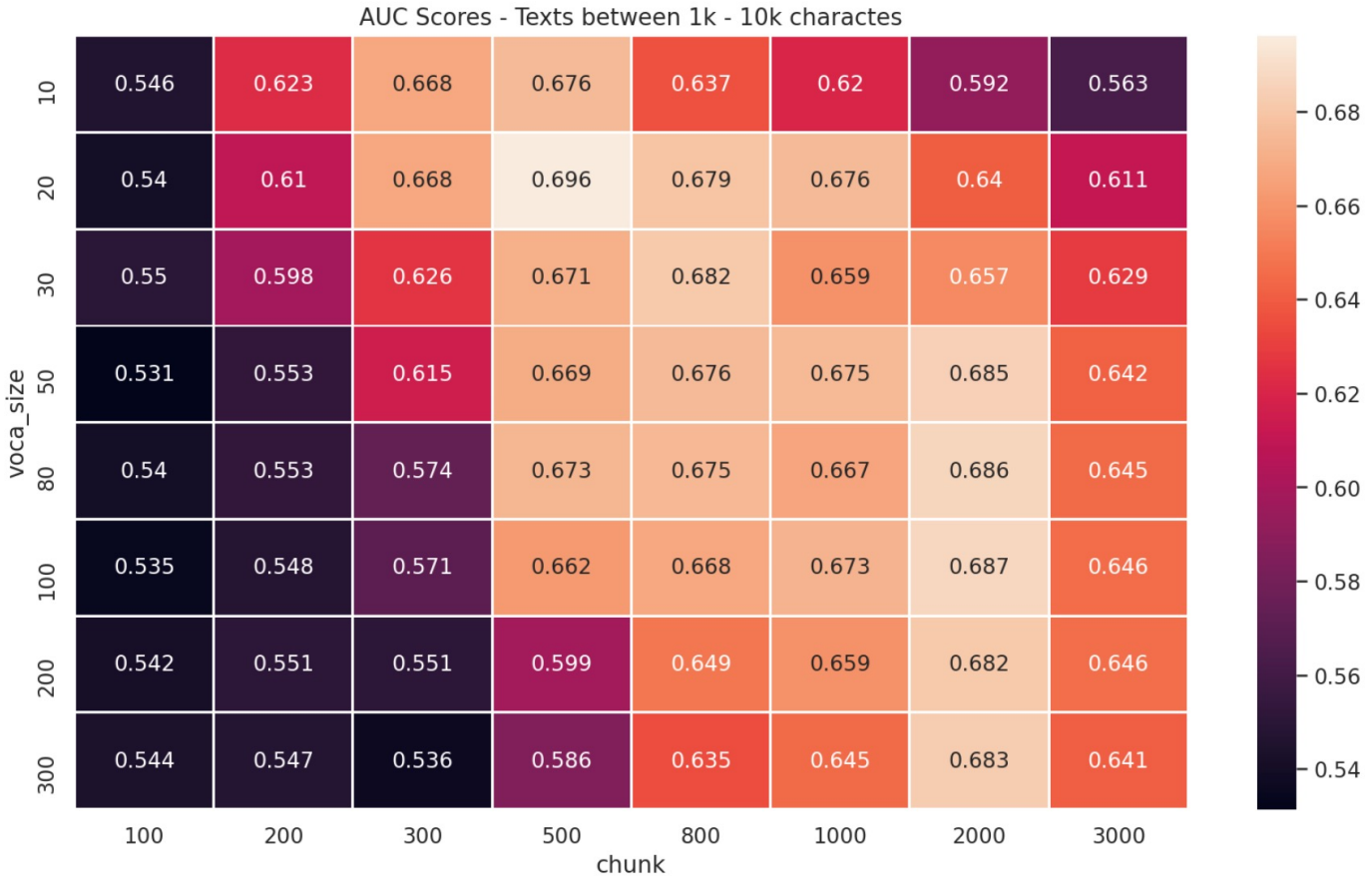
	Intervals	No. intervals	No. pairs of similar authors per interval	No. pairs of different authors per interval	No. pairs per interval	No. pairs
	a	***b***	***c***	***d***	***c+d***	***(c+d)*b***
Short texts (Surveying)	1k-2k, 2k-3k, …, 9k-10k	9	100	100	200	1.800
Long texts (Surveying)	10k-20k, 20k-30k, …, 90k-100k	9	100	100	200	1.800
Mixed-ranged texts (Testing)	1k-2k, 2k-3k, …, 29k-30k	29	200	200	400	11.600

Datasets for surveying and testing ChunkedHCs

4. EXPERIMENTS

4.2. SURVEYING CHUNKEDHCS

Short texts
(Texts between 1,000 & 10,000 characters)

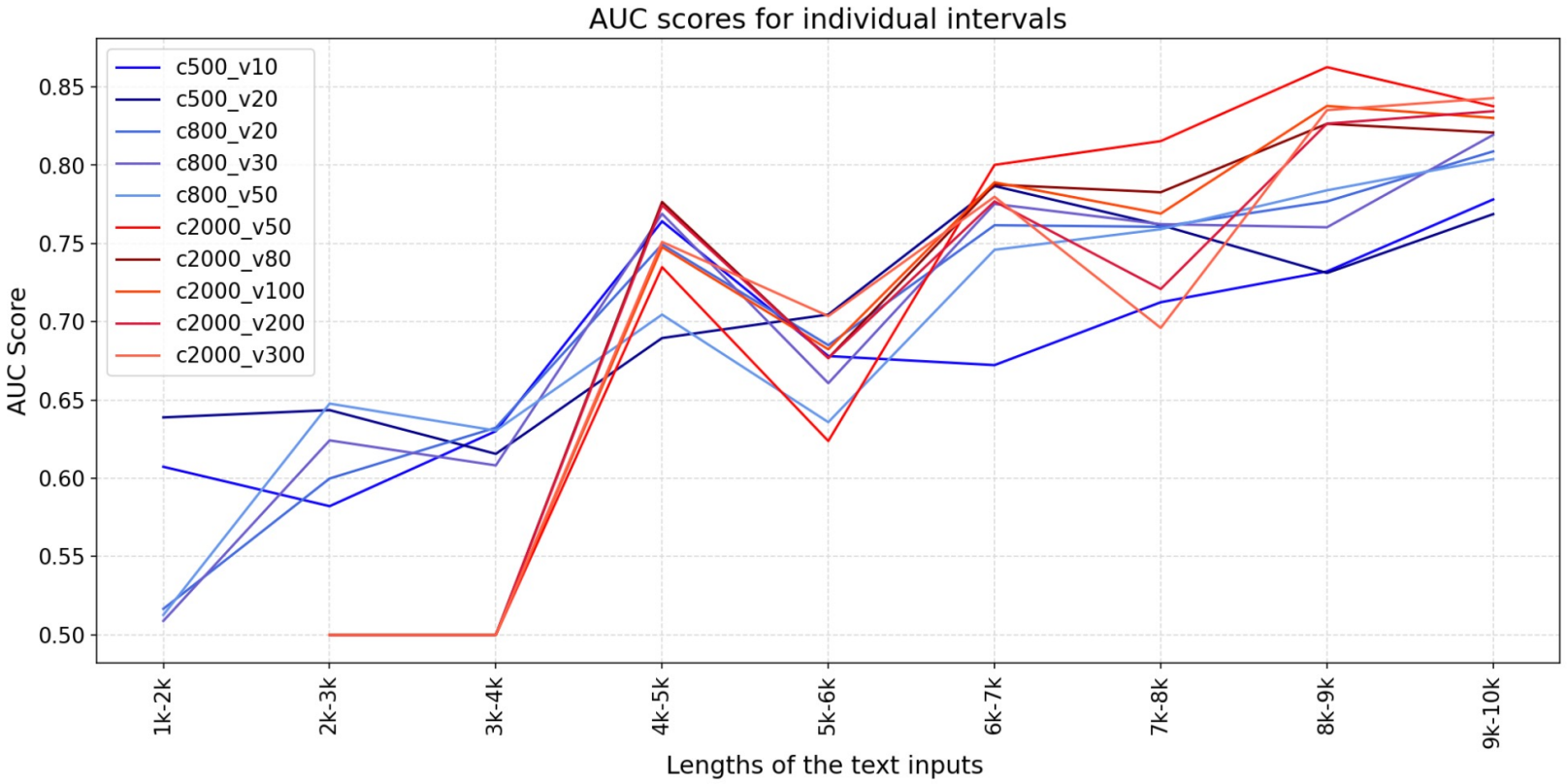


Heatmap – AUC scores with different Chunk sizes and Vocabulary sizes

4. EXPERIMENTS

4.2. SURVEYING CHUNKEDHCS

Short texts
(Texts between 1,000 & 10,000 characters)

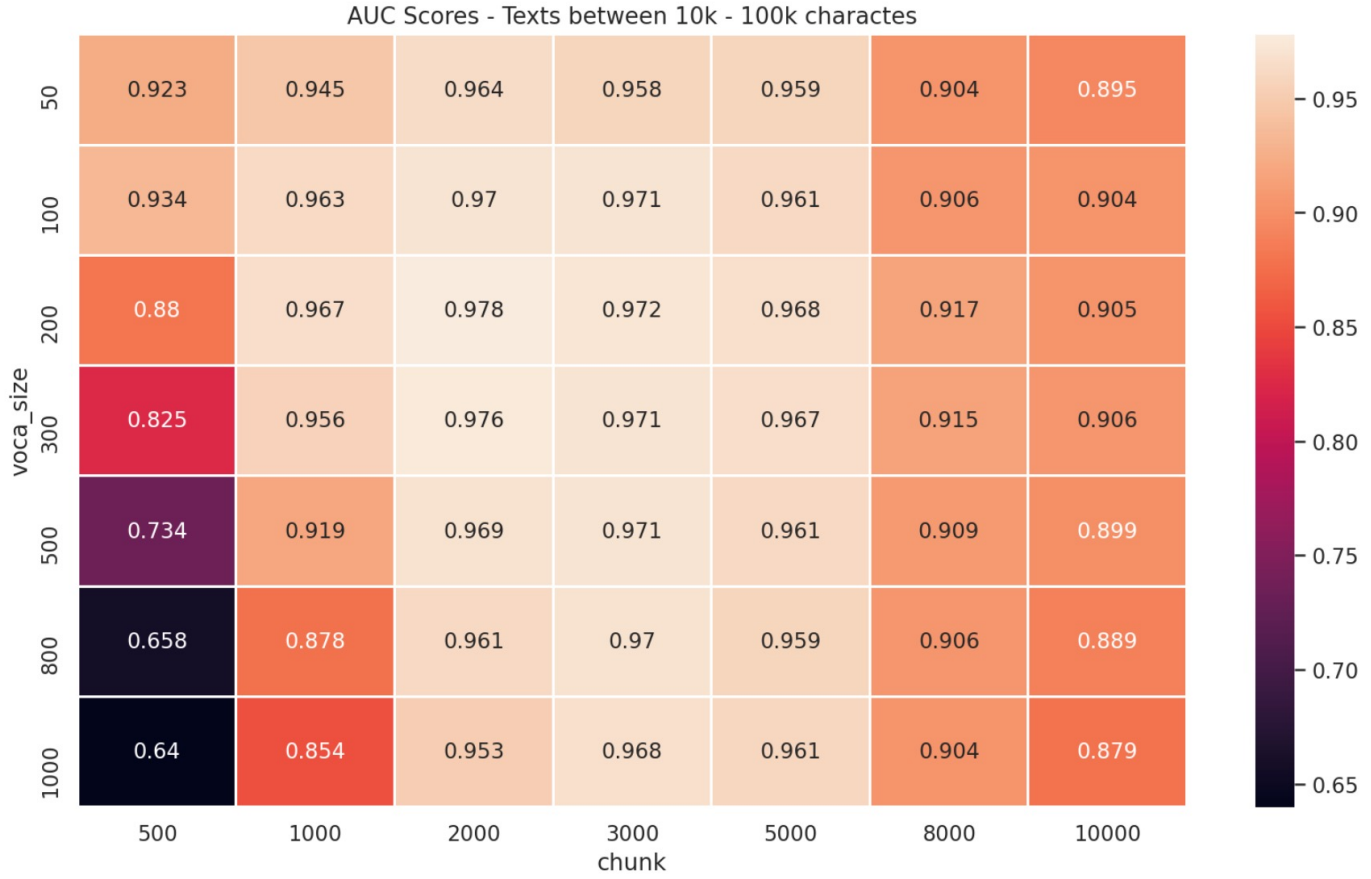


Line plot – AUC scores for individual intervals

4. EXPERIMENTS

4.2. SURVEYING CHUNKEDHCS

Long texts
(Texts between 10,000 & 100,000 characters)

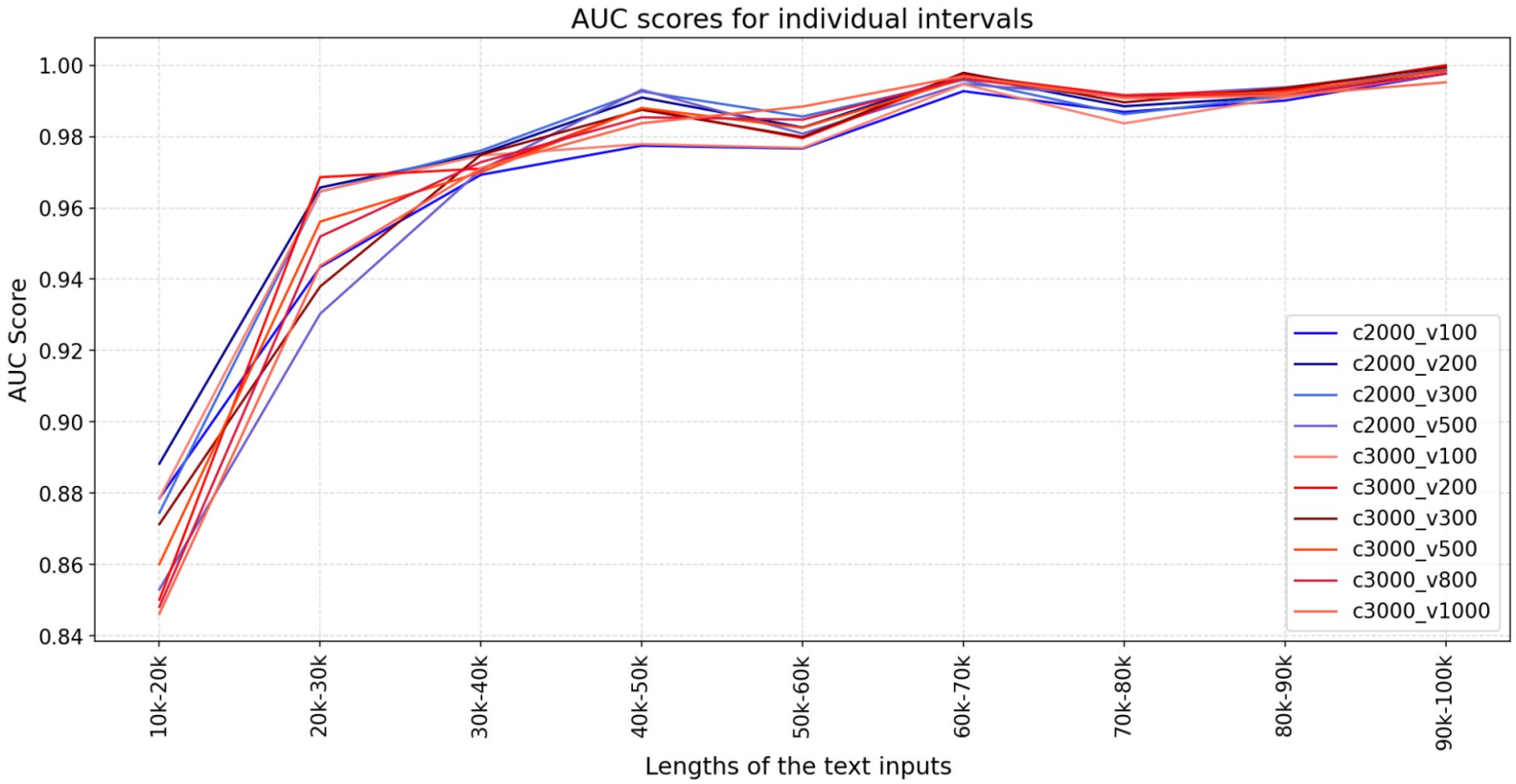


Heatmap – AUC scores with different Chunk sizes and Vocabulary sizes

4. EXPERIMENTS

4.2. SURVEYING CHUNKEDHCS

Long texts
(Texts between 10,000 & 100,000 characters)



Line plot – AUC scores for individual intervals

4. EXPERIMENTS

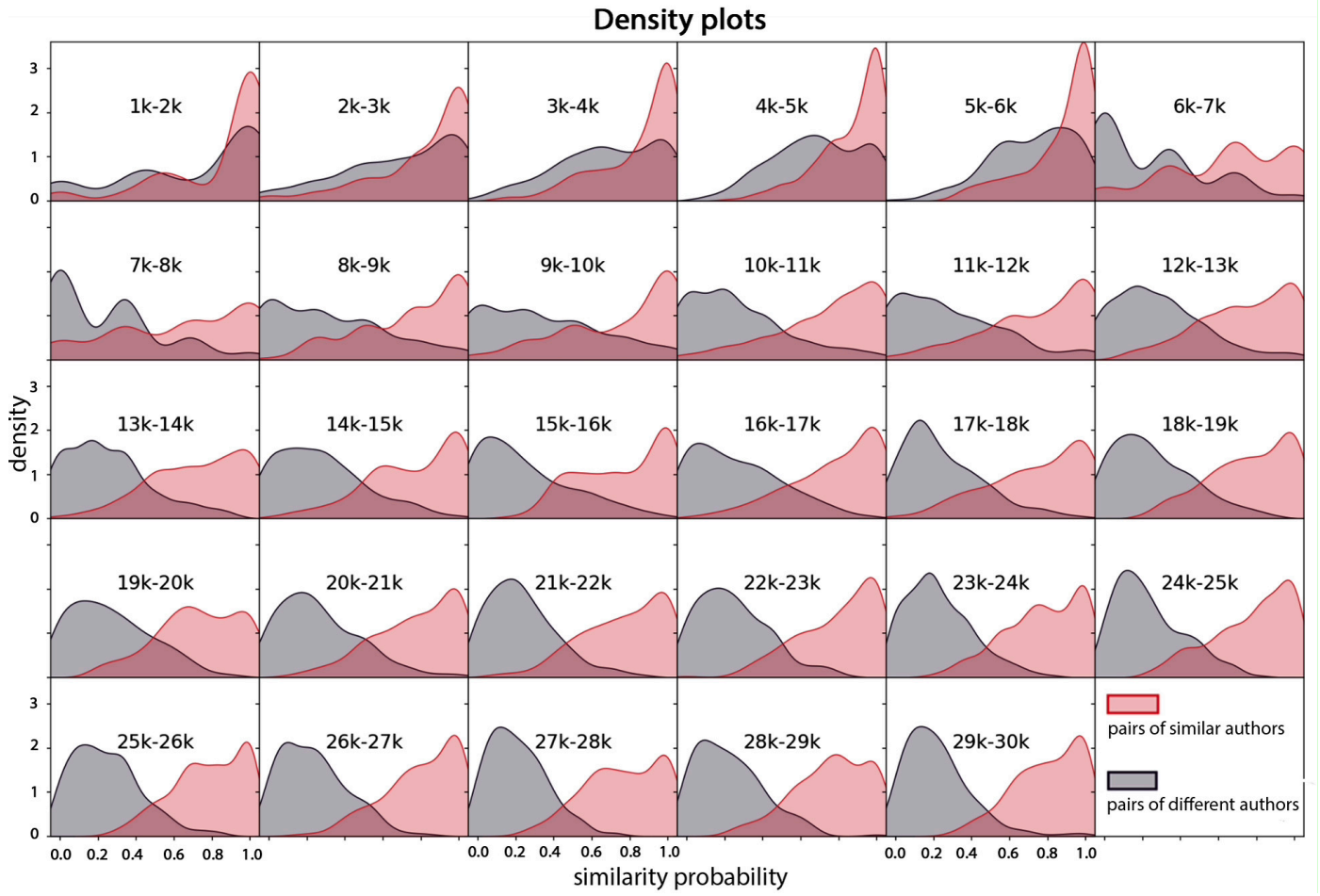
4.3. TESTING CHUNKEDHCS

Intervals	No. Intervals	Chunk size	Vocabulary size
1k-2k, 2k-3k, …, 5k-6k	5	500	20
6k-7k, 7k-8k, …, 9k-10k	4	2.000	100
10k-11k, 11k-12k, …, 29k-30k	20	2.000	200

Choosing Chunk sizes and Vocabulary sizes for the dataset for testing ChunkedHCs

4. EXPERIMENTS

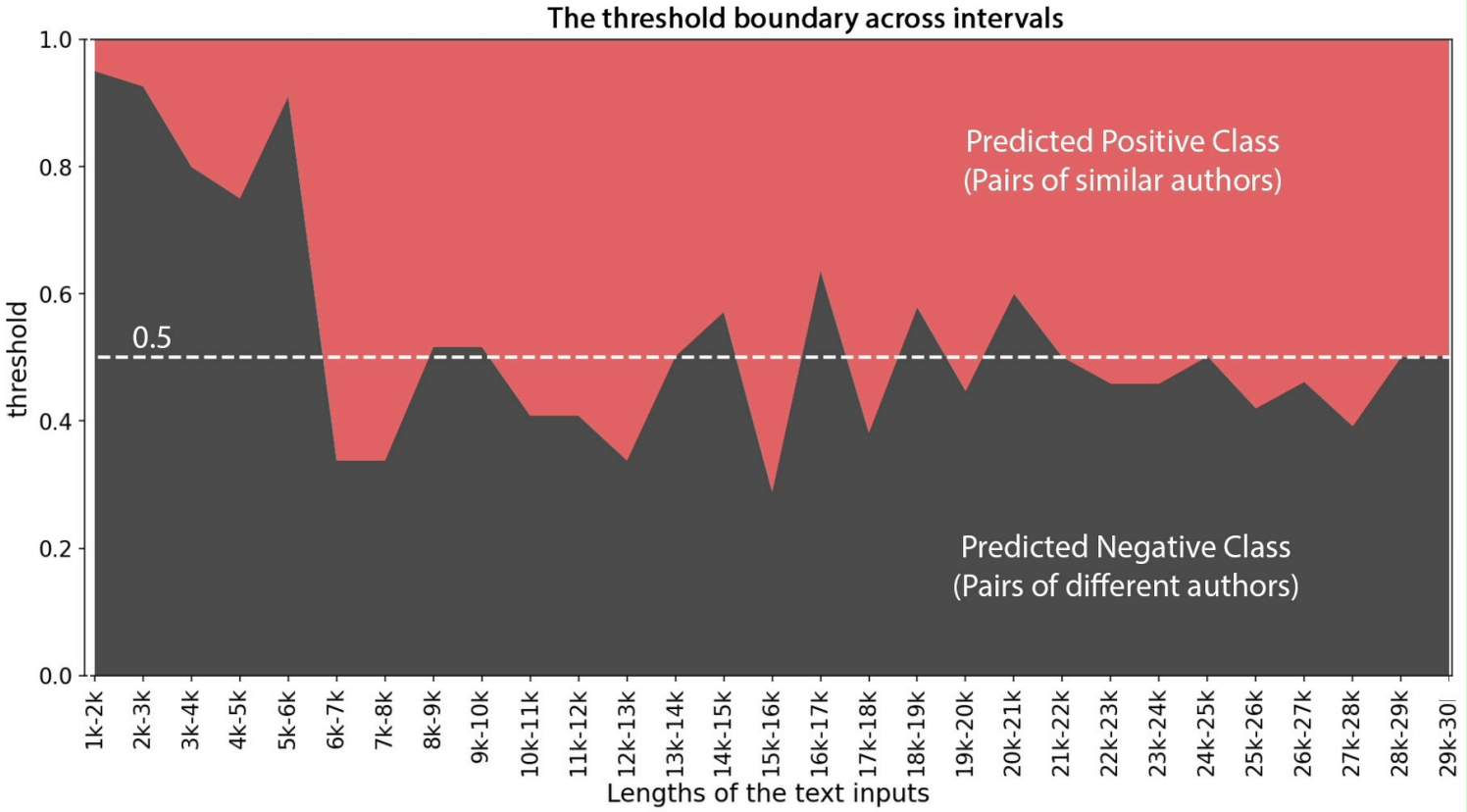
4.3. TESTING CHUNKEDHCS



Density plots – Similarity Probability Distribution

4. EXPERIMENTS

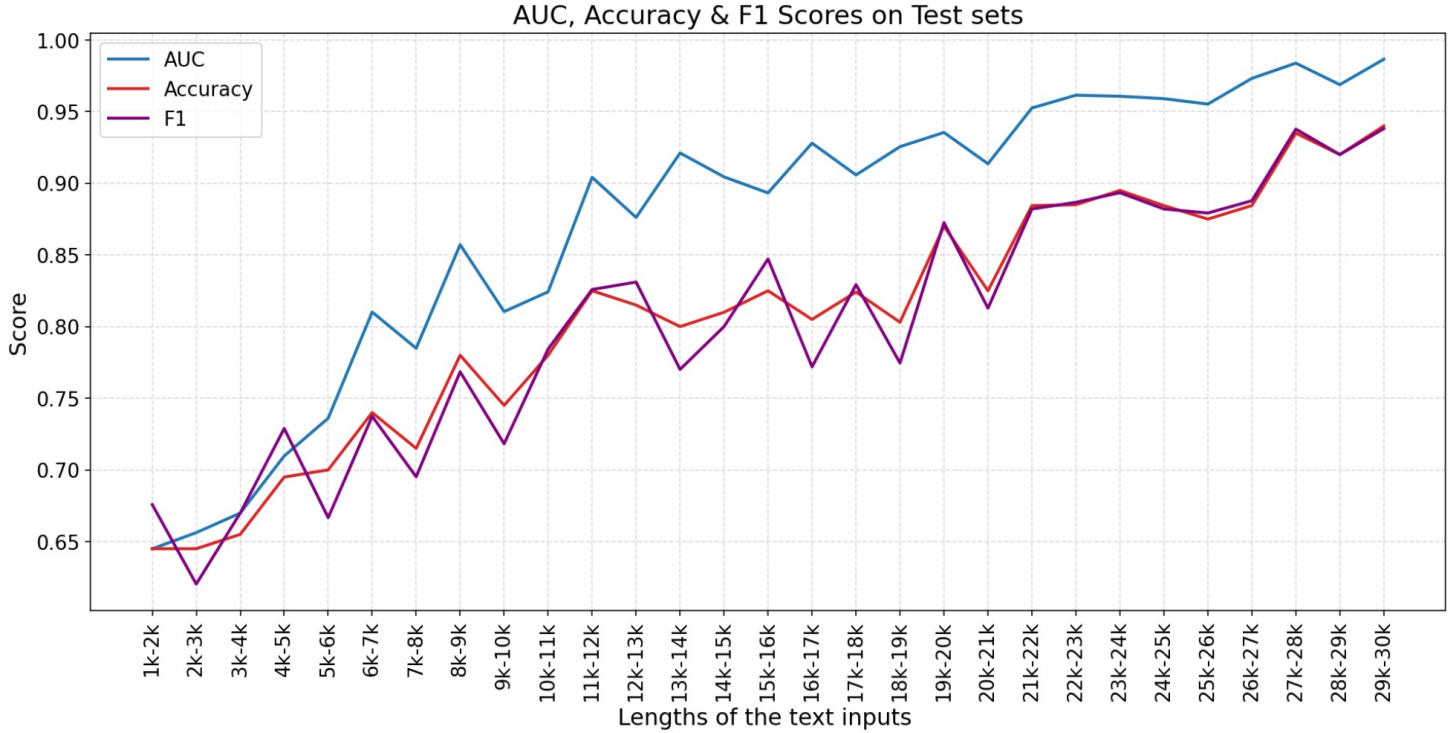
4.3. TESTING CHUNKEDHCS



Choosing thresholds on the validation set
Threshold boundary predicting positive and negative classes

4. EXPERIMENTS

4.3. TESTING CHUNKEDHCS



Range	AUC	Acc.	F1	Range	AUC	Acc.	F1	Range	AUC	Acc.	F1	Range	AUC	Acc.	F1
1k-2k	0.645	0.645	0.6758	**9k-10k**	0.8104	0.745	0.7182	**16k-17k**	0.928	0.805	0.7719	**23k-24k**	0.9607	0.895	0.8934
2k-3k	0.6562	0.645	0.6203	**10k-11k**	0.8242	0.78	0.7843	**17k-18k**	0.9058	0.8241	0.8293	**24k-25k**	0.959	0.8844	0.8821
3k-4k	0.6699	0.655	0.6699	**11k-12k**	0.9041	0.825	0.8259	**18k-19k**	0.9255	0.803	0.7746	**25k-26k**	0.9552	0.875	0.8792
4k-5k	0.7097	0.695	0.7289	**12k-13k**	0.8762	0.815	0.8311	**19k-20k**	0.9354	0.87	0.8725	**26k-27k**	0.9732	0.8844	0.8878
5k-6k	0.7361	0.7	0.6667	**13k-14k**	0.9211	0.8	0.7701	**20k-21k**	0.9136	0.825	0.8128	**27k-28k**	0.9838	0.935	0.9378
6k-7k	0.8101	0.74	0.7374	**14k-15k**	0.9044	0.81	0.8	**21k-22k**	0.9525	0.8844	0.8821	**28k-29k**	0.9688	0.92	0.92
7k-8k	0.7849	0.715	0.6952	**15k-16k**	0.8932	0.825	0.8472	**22k-23k**	0.9614	0.885	0.8867	**29k-30k**	0.9866	0.94	0.9381
8k-9k	0.8572	0.78	0.7684												

AUC, Accuracy and F1 scores on the test set

5. DISCUSSION

5.1. INPUT

- The algorithm does not require sophisticated pre-processing steps.
- ChunkedHCs could be theoretically suitable for other languages than English, and also being suitable for other social media platforms such as Facebook, Twitter, etc.
- ChunkedHCs might not adequately capture an author's writing style due to the Zipf distribution of monograms. The underrepresented features would potentially cost the algorithm the verification ability.
- It is advisable that the class imbalance should be cautiously considered when applying ChunkedHCs. This study of ChunkedHCs used datasets featuring balanced classes, in which the number of text pairs of similar authors is equal to the number of text pairs of different authors.
- It is recommended that there should not be a huge difference between lengths of the two text inputs to avoid bias in the internal binary classification.

5.2. OUTPUT

- ChunkedHCs directly provide a similarity probability,
  - 0 → a pair of different author.
  - 1 → a pair of the same author.
- In most cases, it may be more challenging to conclude with less definitive probability values.
- The biggest obstacle to the authorship verification task is choosing a universal threshold distinguishing the positive/negative classes (Halvani et al., 2016).
- The threshold might need to be adjusted for different datasets and may also depend on other factors.

5. DISCUSSION

5.3. PERFORMANCE

- ChunkedHCs provide good results when text inputs are sufficiently long.
- It is crucial to carefully select chunk size, vocabulary size and decision threshold for different input lengths.
- From the empirical results,
    - For **short texts**, the chunk size should be small and the decision threshold needs to take a more conservative value closer to the upper limit of 1.
    - For **long texts**, the chunk size should be approximately 2,000 to 3,000 characters, with the common threshold of 0.5.

5.4. RUNTIME

- Applying ChunkedHCs is fast and straightforward without having training data phase.
- Measuring the HC statistic has a moderate computational cost, i.e. $O(Nlog(N))$ (Donoho & Jin, 2015).
- ChunkedHCs will be slower with longer texts, or having shorter chunk sizes with larger vocabulary sizes.

5.5. COMPARISON WITH OTHER APPROACHES

- ChunkedHCs offer a much simpler workflow than ML/DL approaches.
- Nevertheless, it might come at the cost of the verification ability, such that ChunkedHCs performance might be not as good as expected.
- This preliminary paper serves as an introduction to ChunkedHCs; as such, the algorithm was not fully compared with other authorship verification approaches.

6. CONCLUSION & FUTURE WORK

- **ChunkedHCs** is an algorithm specifically designed for the authorship verification task to identify whether a pair of texts are written by the same person. It is based on the statistical testing Higher Criticism (Donoho & Jin, 2004) and HC-based similarity algorithm (Kipnis, 2020a & 2020b) (Kestemont et al., 2020).

- Given a pair of texts, **ChunkedHCs** will provide a similarity probability from 0 to 1. The lower limit indicates the pair was written by two different people; meanwhile, the upper limit suggests the same person composed the texts. The algorithm requires careful selection of chunk sizes, vocabulary sizes and decision thresholds for different input lengths. However**, ChunkedHCs** only require simple pre-processing steps such as removing names, numbers and/or punctuations.

- Applying **ChunkedHCs** on the Reddit users' data indicated that the algorithm's performance improves for longer text inputs. **ChunkedHCs** achieve decent classification results for short texts with an accuracy of 0.645 and an F1 of 0.6758 for shortest texts between 1,000 and 2,000 characters. For long texts, **ChunkedHCs** obtain impressive outcomes with an accuracy of up to 0.94 and an F1 of 0.9381 for texts between 29,000 and 30,000 characters.

- **Future work**:
  - Grouping rarer words sharing some similarity together.
  - Mapping all words to their corresponding Part of Speech (POS) tags with some modifications for the text inputs → This shows promising results.
  - Optimising key parameters, including the chunk size, vocabulary size and decision threshold.
  - Expanding the study to other datasets in different languages.
  - Comparing ChunkedHCs performance with other authorship verification approaches.

REFERENCE

Bartoli, A., Dagri, A., Lorenzo, A. D., Medvet, E. and Tarlao, F., 2015. An Author Verification Approach Based on Differential Features – Notebook for PAN at CLEF 2015. In: Cappellato, L., Ferro, N., Jones, G. and Juan, E. S., editors, *CLEF 2015 Evaluation Labs and Workshop – Working Notes Papers, 8-11 September, Toulouse, Frances*. CEUR-WS.org. https://pan.webis.de/downloads/publications/papers/bartoli_2015b.pdf

Boenninghoff, B., Rupp, J., Nickel, R. M. and Kolossa, D., 2020. Deep Bayes Factor Scoring for Authorship Verification – Notebook for PAN at CLEF 2020. In: Cappellato, L., Eickhoff, C., Ferro, N. and Névéol, A., editors, *CLEF 2020 Labs and Workshops, Notebook Papers, 22-25 September, Thessaloniki, Greece*. CEUR-WS.org. https://pan.webis.de/downloads/publications/papers/boenninghoff_2020.pdf

Donoho, D. and Jin, J., 2004. Higher criticism for detecting sparse heterogeneous mixtures. *The Annals of Statistics*, 32(3), pp.962-994. https://arxiv.org/pdf/math/0410072.pdf

Donoho, D. and Jin, J., 2008. Higher criticism thresholding: Optimal feature selection when useful features are rare and weak. *Proceedings of the National Academy of Sciences 105*(39), pp.14790–14795. https://www.pnas.org/content/105/39/14790

Donoho, D. and Jin, J., 2009. Feature selection by higher criticism thresholding achieves the optimal phase diagram. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1906), pp.4449-4470. https://royalsocietypublishing.org/doi/10.1098/rsta.2009.0129#

Donoho, D. and Jin, J., 2015. Higher Criticism for Large-Scale Inference, Especially for Rare and Weak Effects. *Statistical Science*, 30(1), pp.1-25. https://projecteuclid.org/download/pdfview_1/euclid.ss/1425492437

Facebook, 2019. *Facebook Q4 2019 Results* [Online]. Available from https://s21.q4cdn.com/399680738/files/doc_financials/2019/q4/Q4-2019-Earnings-Presentation-_final.pdf [Accessed 26th January 2021].

Frery, J., Largeron, C. and Juganaru-Mathieu, M., 2014. UJM at CLEF in Author Identification – Notebook for PAN at CLEF 2014. In: Cappellato, L., Ferro, N., Halvey, M., and Kraaij, W., editors, *Working Notes Papers of the CLEF 2014 Evaluation Labs, 15-18 September, Sheffield, UK*. CEUR-WS.org. https://pan.webis.de/downloads/publications/papers/frery_2014.pdf

Halvani, O., Steinebach, M. and Zimmermann, R., 2013. Authorship Verification via K-Nearest Neighbor Estimation Notebook for PAN at CLEF 2013. In: Forner, P. Navigli, R. and Tufis, D., editors, *CLEF 2013 Evaluation Labs and Workshop – Working Notes Papers, 23-26 September, Valencia, Spain*. CEUR-WS.org. https://pan.webis.de/downloads/publications/papers/halvani_2013.pdf

Halvani, O., Winter, C. and Pflug, A., 2016. Authorship verification for different languages, genres and topics. *Digital Investigation*, 16, pp.S33-S43. DFRWS EU 2016. https://www.sciencedirect.com/science/article/pii/S1742287616000074

Kestemont, M., Manjavacas, E., Markov, I., Bevendorff, J., Wiegmann, M., Stamatatos, E., Potthast, M., and Stein, B. 2020. Overview of the Cross-Domain Authorship Verification Task at PAN 2020. In: Cappellato, L., Eickhoff, C., Ferro, N. and Névéol, A., editors, *CLEF 2020 Labs and Workshops, Notebook Papers, 22-25 September 2020, Thessaloniki, Greece*. CEUR-WS.org. https://pan.webis.de/downloads/publications/papers/kestemont_2020.pdf

Kipnis, A., 2020a. *Higher Criticism for discriminating word-frequency tables and testing authorship*. arXiv:1911.01208v3 [cs.CL]. https://arxiv.org/pdf/1911.01208.pdf

Kipnis, A., 2020b. Higher Criticism as an Unsupervised Authorship Discriminator – Notebook for PAN at CLEF 2020. In: Cappellato, L., Eickhoff, C., Ferro, N. and Névéol, A., editors, *CLEF 2020 Labs and Workshops, Notebook Papers, 22-25 September 2020, Thessaloniki, Greece*. CEUR-WS.org. https://pan.webis.de/downloads/publications/papers/kipnis_2020.pdf

Stamatatos, S., Daelemans, W., Verhoeven, B., Potthast, M., Stein, B., Juola, P., Sanchez-Perez, M. A. & Barrón-Cedeño, A., 2014. Overview of the Author Identification Task at PAN 2014. In: Cappellato, L., Ferro, N., Halvey, M. & Kraaij, W., editors, *Working Notes Papers of the CLEF 2014 Evaluation Labs, 15-18 September 2014, Sheffield, UK*. CEUR-WS.org CEUR-WS.org. https://pan.webis.de/downloads/publications/papers/stamatatos_2014.pdf

Statista, 2020. *Most popular social networks worldwide as of October 2020, ranked by number of active users* [Online]. Available from https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/ [Accessed 5th January 2021].

THANK YOU FOR LISTENING