Trust in online data : privacy in text, and semantic-based author verification in micro-messages

Khodor Hammoud

Université de Paris

LIPADE
Laboratoire d'Informatique PAris DEscartes

Trust in Online Data: Privacy in Text, and Semantic-Based Author Verification in Micro-Messages

Thèse de doctorat de Université de Paris

École doctorale n°130 EDITE (Sigle)
Spécialité de doctorat: Science des Données

Thèse présentée et soutenue à Paris, 15 November 2021, by

Khodor Hammoud

Composition du Jury :

M. Allel Hadjali Professeur, ISAE ENSMA	Rapporteur
M. Mustapha Lebbah MCU-HDR, Université Sorbonne Paris Nord	Rapporteur
Mme. Valentina Dragos CR-HDR, ONERA	Examinatrice
M. Yücel Saygin Professeur, Sabanci University Turkey	Examinateur
M. Mourad Ouziri MCU, Université de Paris	Invité
Mme. Salima Benbernou Directrice de thèse, Université de Paris	Professeur

Thèse de doctorat

Acknowledgements

I take this opportunity to express my sincere gratitude to my teachers, colleagues, family and friends. It brings me great pleasure to acknowledge everyone who played a helpful or supportive role in the completion of my PhD research.

I would like to thank my supervisors Pr. Salima Benbernou and Dr. Mourad Ouziri from Université de Paris, France, firstly for granting me the chance to pursue a doctoral degree, and secondly for their guidance and constant support throughout my journey. I also want to thank Pr. Yücel Saygin from Sabanci University, Turkey, for the great help and guidance which he provided in the first part of my thesis. My PhD was also partly supervised by Dr. Yehia Taher from the University of Versailles and Dr. Rafiqul Haque from intelligencia to whom I will always be grateful.

Besides my advisors, I would like to thank the rest of my thesis committee: Professor Allel Hadjali, from École Nationale Supérieure de Mécanique et d'Aérotechnique, MCU-HDR Mustapha Lebbah, from Université Sorbonne Paris Nord, and CR-HDR Valentina Dragos from the Office National d'Etudes et de Recherches Aérospatiales, for their time and efforts in reviewing my thesis, and also their insightful comments and constructive feedback.

I am incredibly grateful to my colleagues of the LIPADE group, the presence of whom made my research path easier and more fun. With them, I've exchanged experiences and knowledge which made the research process more interesting and interactive. I also want to thank my dear friend María Camila Esquivel Ortegon for the great times and continuous support; without whom, the year 2020 would have been all that much more difficult.

Finally, I want to thank my family for the continuous support. Through the tough times that I've been through, that they've been through, and that our country Lebanon has been through, they always remained fully supportive throughout. My mother Sanaa Hassan, father Souheil Hammoud, sisters Fatima Hammoud, Rasha Hammoud and Nour Hammoud, my brother-in-law Ali Saad, and last but certainly not least, my friend, colleague, and brother-in-law Dr. Nabil Abdoun. It would have not been possible without all of you.

Publications

Published

Khodor Hammoud, Salima Benbernou, Mourad Ouziri, Yucel Saygin, Rafiqul Haque, Yehia Taher,**"Personal Information Privacy: What's Next?"**. The 2^{nd} International Conference on Big Data and Cyber-Security Intelligence (BDCSINTELL) 2019.

Khodor Hammoud, Salima Benbernou, Mourad Ouziri, **"A Sentiment-Based Author Verification Model Against Social Media Fraud"**, The 12^{th} Conference of the European Society for Fuzzy Logic and Technology, (IFSA-EUSFLAT) 2021.

In Progress

Khodor Hammoud, Salima Benbernou, Mourad Ouziri, **"Opinion-Based Author Verification in Micro Messages"**.

Khodor Hammoud, Salima Benbernou, Mourad Ouziri, **"Temporal Effect on Opinion-based Author Verification in Micro-Messages"**.

Abstract

Many Problems surround the spread and use of data on social media. There is a need to promote trust on social platforms, regarding the sharing and consumption of data. Data online is mostly in textual form which poses challenges for automation solutions because of the richness of natural language. In addition, the use of micro-messages as the main means of communication on social media makes the problem much more challenging because of the scarceness of features to analyze per body of text. Our experiments show that data anonymity solutions cannot preserve user anonymity without sacrificing data quality. In addition, in the field of author verification, which is the problem of determining if a body of text was written by a specific person or not, given a set of documents known to be authored by them, we found a lack of research working with micro-messages. We also noticed that the state-of-the-art does not take text semantics into consideration, making them vulnerable to impersonation attacks.

Motivated by these findings, we devote this thesis to tackle the tasks of (1) identifying the current problems with user data anonymity in text, and provide an initial novel semantic-based approach to tackle this problem, (2) study author verification in micro-messages and identify the challenges in this field, and develop a novel semantics-based approach to solve these challenges, and (3) study the effect of including semantics in handling manipulation attacks, and the temporal effect of data, where the authors might have changing opinions over time.

The first part of the thesis focuses on user anonymity in textual data, with the aim to anonymize personal information from online user data for safe data analysis without compromising users' privacy. We present an initial novel semantic-based approach, which can be customized to balance between preserving data quality and maximizing user anonymity depending on the application at hand.

In the second part, we study author verification in micro-messages on social media. We confirm the lack of research in author verification on micro-messages, and we show that the state-of-the-art, which primarily handles long and medium-sized texts, does not perform well when applied on micro-messages. Then we present a semantics-based novel approach which uses word embeddings and sentiment analysis to collect the author's opinion history to determine the correctness of the claim of authorship, and show its competitive performance on micro-messages.

We use these results in the third part of the thesis to further improve upon our approach. We construct a dataset consisting of the tweets of the 88 most followed twitter influencers. We use it to show that the state-of-the-art is not able to handle impersonation attacks, where the content of a tweet is altered, changing the message behind the tweet, while the writing pattern is preserved. On the other hand, since our approach is aware of the text's semantics, it is able to detect text manipulations with an accuracy above 90%.

And in the fourth part of the thesis, we analyze the temporal effect of data on our approach for author verification. We study the change of authors' opinions over time, and how to accommodate for that in our approach. We study trends of sentiments of an author per a specific topic over a period of time, and predict false authorship claims depending on what timeframe does the claim of authorship fall in.

Résumé

De nombreux problèmes entourent la diffusion et l'utilisation des données sur les réseaux sociaux. Le partage d'informations sur les réseaux sociaux est une composante essentielle des interactions en ligne, où chaque action qu'un utilisateur effectue en ligne contribué à une certaine forme de partage d'informations. Un élément essentiel sous-jacent à tout partage d'informations en ligne est la confiance des utilisateurs. Il est nécessaire de promouvoir la confiance sur les plateformes sociales, en ce qui concerne le partage et la consommation de données. Cela ouvre la porte à des questions telles que : avec qui partager les informations, combien d'informations partager, à quelles sources d'information faire confiance et comment confirmer la vérité derrière l'identité des sources d'information ; une fausse source d'information est susceptible de produire de fausses informations. Les données en ligne sont principalement sous forme textuelle, ce qui pose des problèmes aux solutions d'automatisation en raison de la richesse du langage naturel. De plus, l'utilisation des micro-messages comme principal moyen de communication sur les médias sociaux rend le problème beaucoup plus délicat en raison de la rareté des fonctionnalités à analyser par corps de texte. Nos expériences montrent que les solutions d'anonymat des données ne peuvent pas préserver l'anonymat des utilisateurs sans sacrifier la qualité des données. De plus, dans le domaine de la vérification d'auteur, qui est le problème de déterminer si un corps de texte a été écrit par une personne spécifique ou non, étant donné un ensemble de documents dont l'auteur est connu, nous avons constaté un manque de recherche travaillant avec micro-messages. Nous avons également remarqué que l'état de l'art ne prend pas en considération la sémantique des textes, ce qui les rend vulnérables aux attaques par usurpation d'identité. De plus, nous avons une métrique de confiance produite par le modèle.

Motivés par ces résultats, nous consacrons cette thèse pour aborder les tâches de (1) identifier les problèmes actuels avec l'anonymat des données utilisateur dans le texte, et fournir une première approche sémantique originale pour résoudre ce problème, (2) étudier la vérification de l'auteur en micro -messages et identifier les défis dans ce domaine, et développer une nouvelle approche basée sur la sémantique pour résoudre ces défis, et (3) étudier l'effet de l'inclusion de la sémantique dans la gestion des attaques de manipulation, et l'effet temporel des données, où les auteurs pourraient avoir changé les opinions au fil du temps.

La première partie de la thèse se concentre sur l'anonymat des utilisateurs dans les données textuelles, dans le but d'anonymiser les informations personnelles des données des utilisateurs en ligne pour une analyse sécurisée des données sans compromettre la confidentialité des utilisateurs. Nous présentons une première approche basée sur la sémantique, qui peut être personnalisée pour équilibrer la préservation de la qualité des données et la maximisation de l'anonymat de l'utilisateur en fonction de l'application à portée de main. Pour qu'un document en question pose un problème de confidentialité, il doit contenir suffisamment d'informations d'identification pour identifier de manière unique la personne qui lui est associée, et doit contenir des informations privées ou sensibles. Si l'une de ces con-

ditions n'est pas remplie, le document ne provoquera pas de fuites de confidentialité. Par conséquent, le degré de risques pour la vie privée associée à un document est une combinaison des informations d'identification et des informations privées qu'il contient. Notre objectif est de : (1) Fournir une métrique pour évaluer le degré d'identifiabilité d'un document textuel donné. C'est-à-dire. dans quelle mesure la personne dont les informations sont présentes dans le document peut-elle être réidentifiée. (2) Fournir une métrique pour évaluer la sensibilité des informations privées contenues dans ledit document. (3) Fournir une méthodologie pour anonymiser la personne mentionnée dans le document afin que le degré de risque de ré identification soit inférieur à un certain seuil. Nous définissons 2 nouveaux concepts orientés métier : $SEED_I$, la liste des termes identifiants dans un domaine métier donné, et $SEED_S$, la liste des termes sensibles dans un domaine métier donné.

Dans la deuxième partie, nous étudions la vérification d'auteur dans les micro-messages sur les réseaux sociaux. Nous confirmons le manque de recherche en vérification d'auteur sur les micro-messages, et nous montrons que l'état de l'art, qui traite principalement des textes longs et moyens, ne fonctionne pas bien lorsqu'il est appliqué sur des micro-messages. En fait, nous testons l'un des modèles les plus performants pour la vérification d'auteur en texte long, et nous remarquons une baisse d'environ 50% des performances lors de l'expérimentation de micro-messages. Ensuite, nous présentons une nouvelle approche basée sur la sémantique qui utilise des inclusions de mots et une analyse des sentiments pour collecter l'historique des opinions de l'auteur afin de déterminer l'exactitude de la revendication de paternité et montrer ses performances concurrentielles sur les micro-messages. Étant donné un document d_x prétendant être publié par un auteur A, nous extrayons les pairs sujets-opinion des documents connus pour être publiés par A, que nous appelons l'historique des opinions de l'auteur, qui ont un sujet similaire à d_x, et comparez les sentiments des documents extraits à ce dé d_x. Si nous trouvons une contradiction dans les opinions, alors il serait peu probable que d_x ait été publié par A. Notre modèle fournit également un score de confiance, qui est une estimation de ses performances pour prédire si A est le véritable auteur d'un document dont la paternité est remise en question d_x. Le calcul de la confiance dépend de la similitude entre les incorporations de mots de chacun des sujets les plus similaires dans l'historique des opinions et l'incorporation de mots du sujet dé d_x ; le document de paternité contestée. Plus la similarité entre ces plongements est élevée, plus la confiance de la prédiction du modèle est élevée. En conséquence, notre modèle produit des résultats d'une grande précision en ne choisissant que des documents de grande similarité de sujet avec d_x. Cela implique que, étant donné suffisamment de documents d'une grande pertinence pour un sujet, nous capturons effectivement l'opinion d'un auteur A sur ce sujet. L'implication était d'acquérir suffisamment de documents centrés sur ce sujet pour prendre une décision précise.

Nous utilisons ces résultats dans la troisième partie de la thèse pour améliorer encore notre approche. Nous appliquons une technique de clustering aux sentiments pour la vérification de la véritable paternité et introduisons les différentes manières dont nous réglons

les hypers paramètres. Dans cette technique, nous modélisons le problème de vérification d'auteur comme un problème de détection d'anomalies, où nous vérifions les sentiments des documents de paternité interrogés pour voir s'ils entrent dans les normes des A pour un sujet donné, ou s'ils s'enregistrent comme une anomalie . Nous utilisons DBSCAN, un algorithme de clustering, comme outil de détection d'anomalies. Il regroupe les points de données en groupes, et les points qui ne rentrent pas dans un groupe (anomalies) sont étiquetés comme -1. Nous distinguons deux approches différentes pour régler les différents hypers paramètres impliqués dans notre modèle : $Static$ et $Dynamic$, où le réglage statique est un réglage manuel de chaque paramètre, tandis que dans le cas du réglage dynamique, nous laissons les paramètres s'adapter à au nombre de documents obtenus à partir de l'historique des opinions de l'auteur qui correspondent au sujet de d_x.

Nous introduisons également un nouvel ensemble de données d'influenceurs Twitter, que nous avons collecté dans le but de tester notre modèle. Nous développons un nouvel ensemble de données Twitter formé des tweets les plus récents des 88 utilisateurs Twitter les plus suivis et récupérons les 3200 tweets les plus récents par auteur. Nous partons de l'idée qu'un imitateur serait plus susceptible de se faire passer pour un influenceur en ligne qu'un utilisateur avec une faible présence en ligne.

De plus, nous testons Adominem, le plus performant de la tâche de vérification d'auteur PAN 2020, qui se concentre sur la vérification d'auteur dans un texte court, et montrons que ses performances chutent de 33% lorsqu'elles sont appliquées aux micro-messages. C'est un excellent candidat pour tester les micro-messages, car il a été spécialement conçu pour s'attaquer à la vérification de l'auteur dans un texte court. Enfin, nous créons à la main un ensemble de documents où nous modifions la sémantique du document, tout en conservant le style d'écriture. Il s'agit ici de montrer un point faible présent dans les approches actuelles de vérification d'auteur, à savoir qu'elles ne prennent pas en compte la sémantique, ce qui les rend vulnérables aux documents manipulés qui préservent les caractéristiques stylistiques de l'auteur. Nous effectuons ces tests à la fois sur notre modèle et sur l'état de l'art Adhominem à des fins de comparaison. Notre modèle reçoit une précision de 93%, tandis qu'Adhominem atteint une précision de 36%.

Et dans la quatrième partie de la thèse, nous analysons l'effet temporel des données sur notre approche de vérification d'auteur. Nous étudions l'évolution des opinions des auteurs au fil du temps et comment s'en accommoder dans notre approche. Un problème avec notre approche initiale est que les gens ont tendance à changer d'opinion avec le temps. Un auteur bien connu pour aimer les téléphones Apple pourrait changer d'avis si Apple sortait un mauvais téléphone ou si cet auteur découvrait un téléphone qu'il aime et qui n'a pas été fabriqué par Apple. Ainsi, il est important d'étudier l'effet temporel des données sur la vérification de la paternité. C'est un défi car chaque auteur aurait son propre rythme de variation d'opinion au fil du temps, et dans l'historique d'opinion d'un auteur, différents sujets auraient des variations d'opinion différentes au fil du temps. Nous commençons par

développer un ensemble de données artificiel d'opinions où nous supposons que l'ensemble de données entier appartient à un seul auteur et fait référence à un seul sujet. Nous prenons également en compte les évolutions progressives des opinions au fil du temps. Ceci pour refléter le fait que les sentiments des tweets ne feront pas un saut soudain et drastique, par exemple, d'être négatif à être positif. Nous utilisons cet ensemble de données pour montrer que ne pas inclure l'aspect temporel montre une baisse significative des performances. Nous modélisons toujours ce problème comme un problème de clustering, mais étant donné d_x, nous ne considérons plus l'ensemble de données entier pour l'analyse, ou du moins, tous les documents de l'auteur ayant un sujet similaire à d_x. Au lieu de cela, nous prenons un sous-ensemble où les documents ont un sujet similaire à d_x, et tombent également dans une certaine fenêtre de temps à partir de d_x. Nous intégrons la variation d'opinion (écart type) et la densité des points de données dans la fenêtre temporelle dans le réglage des paramètres de l'algorithme de clustering DBSCAN, ce qui nous donne de meilleures performances du modèle. Enfin, nous menons notre expérience sur un sous-ensemble de notre ensemble de données d'influenceurs Twitter, en extrayant un sous-ensemble de tweets qui tournent autour du sujet **Apple**, la société de technologie, puis en insérant un nouveau document qui pourrait provenir de l'auteur eux-mêmes, ou était d'origine fabriquée.

Nos résultats montrent que l'inclusion de l'aspect temporel, lorsqu'il s'agit de vérification d'auteur, joue un rôle important sur la performance. En ajustant l'algorithme de clustering en tenant compte de la densité des opinions et de leur répartition de la polarité, nous produisons des performances élevées pour déterminer les documents de paternité vraie et fausse sur un sous-ensemble de données de l'ensemble de données Twitter de nos auteurs. Une expérimentation supplémentaire doit être appliquée pour évaluer correctement le modèle, avec plus d'auteurs, de données par auteur et de sujets. Cependant, sur la base de nos résultats, nous soutenons que ne pas inclure l'aspect temporel affecterait négativement les performances du modèle, car il nous fournit une représentation plus actuelle de l'opinion d'un auteur au fil du temps.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Information sharing on social networks is an essential component of online interactions, where every action a user takes online contributes to some form of information sharing. An essential component underlying all online information sharing is user trust. User trust on social networks is a topic of interest among the computer science community. It opens the door for questions like: with whom to share information, how much information to share, which information sources could be trusted, and how to confirm the truth behind the identities of information sources; a fake information source is likely to produce fake information. In this regard, there are 2 players at hand:

- the users, who are the people that use social networks, and share their own personal and non-personal information. These users may also be influencers, whom are the online figures whose actions can have an effect on people's lives and decisions.
- the data holders, who are the entities interested in collecting user information available on social media, for a multitude of purposes like targeted advertisement and improving services.

So it's logical that there exists an underlying layer of trust within users and social networks. This is inferred from the fact that people do indeed share their information online. However, there are 2 layers of responsibility here. It is the users' responsibility to firstly make sure not to share too much information online, whether it being personal or sensitive information, and secondly to only trust verified credible information sources. On the other hand, it's the data holders' responsibility to preserve users' privacy when handling their information, whether it being through data analysis or sharing with third parties. With that in mind, it would be beneficiary to have automated systems in place to ensure critical points like user privacy preservation, and data source validation, since it is not difficult to abuse user trust, and take advantage of it:

- One can collect the personal information which are spread on social media to perform actions that violate the privacy of users, like influencing their political decisions [38].
- A data-holding corporation can violate its users' privacy by collecting and analyzing their personal information [101].
- One can use social media to spread fake news [67, 36] with the aim of aiding a certain agenda.
- One can spread content of questionable truth that serve a certain personal benefit.

In order for social media users to gain others' trust over a long time period, they share more information about themselves, allowing access to a lot of personal information to others. Therefore, the more users use social media to get to know others, the more they may trust them. This can have the negative backlash of creating a false sense of security towards social media as a whole, promoting trust towards potentially non-trust-worthy sources, especially with the more-impressionable individuals. This also promotes sharing more information online without double-checking security measures like privacy policies. In [105], the authors found that younger (twenty years of age and younger), female and heavy users of social media are more likely to trust the content they face online than males and older individuals. They have belief in the competence of others, and think people are genuinely concerned about individuals in their network. These points serve to highlight the importance of having solutions that can provide more secure and trust-worthy manners of sharing information online. In our work, we focus on the issues of the protection of user privacy, and information source validation with textual data on social media.

1.1 User Data Privacy

1.1.1 Background

Data is the main asset of enterprises, exploitation of which is currently being addressed in the context of Big Data. A major portion of the data generated by people and collected by enterprises is in the form of unstructured text documents such as tweets, comments and blog posts. People express their opinion, complaints, joy, etc through posts, while companies are hungry to enrich their customer relationship management (CRM) systems with more information about their customers, or they would like to enhance their products and services with respect to the implicit and explicit feedback from the consumers. Let us consider a call-center use case where there is a large scale call center service provider for multiple businesses. Call centers receive information from customers through various means, but in this work we will concentrate on textual format such as emails. The task of a call center in this case is to record and address the problem indicated by the customers via emails. The emails provide valuable information for the industry and service provider. For example, through text analytics, general problems regarding a product can be discovered, or the most frequently complained services can be identified. A Call Center, however, is not an expert on such analytical services, and has to outsource that task to a third party expert specialized on text analytics. Figure 1.1 explains the general architecture where a data holder (call center in our case) needs third party services.

Privacy concerns come into play when any data regarding a real person need to be analyzed by third parties and Data protection regulation applies to our call center text analytics scenario. Proper privacy management needs to be implemented together with an interactive privacy recommendation methodology to intercept and control the data flow between the data holder and the data analyst. A possible solution for privacy protection is de-identification of the data, and/or removal of sensitive information before it can be published or shared with third parties for analytics. However, it has been established that any kind of data release leads to sensitive information leak. Leaked sensitive information may cause privacy breaches, or the compromise of secret sensitive company information. For example, companies which are obtaining the call service may not want others to know about the specific customer problems faced by that company.

Another more specific scenario/use-case is online health services which are a perfect example for demonstrating the privacy risks associated with cloud-based services. Online health services are becoming popular as people seek immediate online advise regarding their health conditions which requires detailed and mostly sensitive personal information to be transferred and stored over the cloud as depicted in Figure 1.2. The data provided by

the patients is again unstructured and mostly in textual format. Since the data is collected and stored in the cloud, we are not even sure where it is stored and how it is going to be used unless specific complex consent forms are provided and agreed upon by the user.

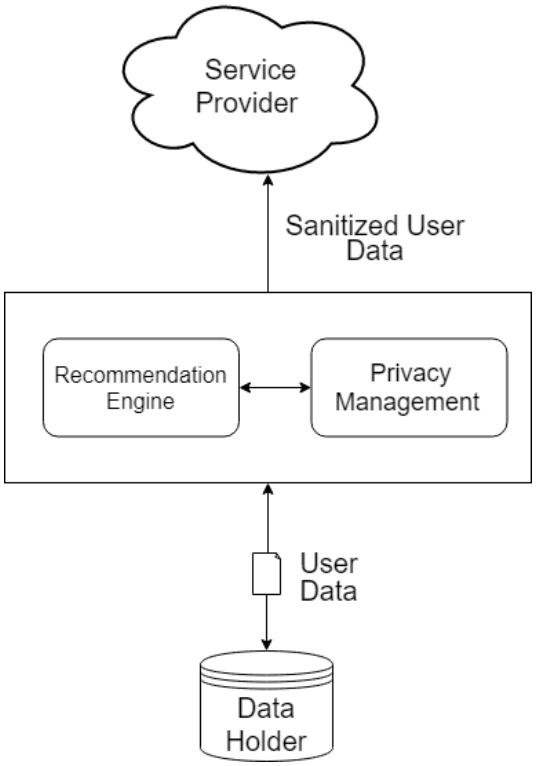

Figure 1.1: General Architecture With Data from Data Holders.

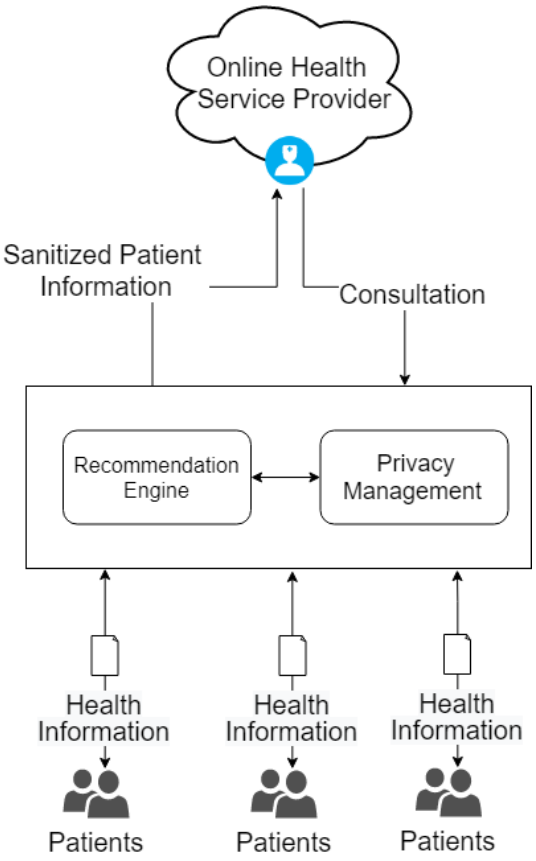

Figure 1.2: General Architecture With User Data.

In addition, a major concern regarding user privacy is the open data movement [76], in which there is a call for governmental data to be available to anyone with a possibility of redistribution in any form without any copyright restriction. The concept behind open data is having more data made available in a useful manner, without any restrictions on usage applied by publishers or data providers, and publication of data that is difficult to reuse, because of poor annotations or un-editable formats, for example, which would drastically help in the betterment of the research and scientific community. The presence of openly available data would result in a greater level of transparency and reproducibility of scientific work, and hence improve efficiency of the scientific process.

There are regulations that govern user data handling, latest of which is the European's General Data Protection Regulation (GDPR[1]), which needs transparency, and user-anonymity when performing statistical analysis, and places heavy fines on violating parties. However, this requires high efforts for providing solutions that adapt to different situations, as there are many formats that data is saved in, and many ways that data is analyzed with. Securing stored/published data depends on the way data is stored. In the past, information was almost strictly stored in the form of structured relational databases [72]. Consequently, shared data was in the form of structured datasets. Ensuring privacy to these datasets was first in the form of deleting the unique identifiers, but then L. Sweeney [99] published a research result that proved that users can still be identified from their quazi-identifiers, and proposed a new methodology known as k-anonymity [84]. Following K-anonymity, several solutions were proposed including ℓ-diversity [71] and t-closeness [69] that address shortcomings discovered in k-anonymity. However, in 2006, Dwork and Aaron introduced differential privacy [26] as a solution for privacy-preserving data analysis which can be used to provide security for both data storage and data analysis.

Further research needs to be conducted on privacy issues in textual data in order to understand the privacy risks in online services and to take appropriate measures to protect the individuals which is the main objective of this work. In the scenarios described above, for privacy preserving text data management, one method is to publish sanitized/anonymized data, and the other is to keep the data private and do privacy preserving text analytics through differential privacy. Here the utility of the data, and the privacy of individuals need to be balanced in order to have reasonable results that could benefit all the stakeholders.

1.1.2 Motivation

The digital transformation created an ample of opportunities for various organizations and adversaries to abuse privacy since the digital systems enable them to hold information of people forever. Despite considerable attention, Web privacy continues to pose significant threats and challenges. One major step is securing the way companies store, share and publish user information, as data regulations impose data publication, which if not secured, can be used to re-identify the individual owners. Organizations such as Google can profile anyone without the users being aware of it. The concrete evidence of personal information abuse are the two recent incidents:

1. The Facebook trail [101] which addresses Facebook exposing the data of up to 87

[1]https://eugdpr.org

million Facebook users to a researcher who worked at Cambridge Analytica. This law suit cost Facebook between 3 billion and 5 billion US dollars with the US Federal Trade Commission (FTC).

2. The Google testimony [39], where Google was sued in a proposed class action, where it was accused of illegally invading the privacy of millions of users by pervasively tracking their internet use through browsers.

These incidents, amongst others, show a clear indication that there are persistent violations to user privacy when it comes to user data analysis. Its not only the software vendors; hardware companies violate the privacy of their users as well. As an example, Samsung's smart TVs potentially spying on their users where the TV appears to be turned off when, in fact, it is not, which allows it to record audio [53] without the knowledge of its users. There are also the recent events revealing the potential of the giant Chinese tech company Huawei using its mobile phone to spy on users, which ended up with the blockage of its phones from using google services and banning them from the US [103]. Hence, users are interested in finding a viable permanent solution to the privacy issue to guarantee the safeguarding of their data online, and have a peace of mind when utilizing online services. On the other hand, corporations and vendors collecting and utilizing user information would benefit from the added user trust, since users would be more likely to utilize their services, in addition to avoiding all the costly law suits and prosecutions that come with privacy breaches. This is a point of focus in Europe, with the GDPR being in application, which permits the EU's Data Protection Authorities to issue fines of up to €20 million euros (24.1 million US dollars) or 4% of annual global turnover (whichever is higher) to any company that violates the GDPR regulations for user privacy.

The Challenges of Privacy Preservation in Text

Textual data is widely diverse in its nature. It comes from different sources such as social media, emails, chats, web pages etc, which creates challenges on the processing requirements, and the mechanisms by which to analyze. Works of privacy in text have been mainly focused on the medical field [22, 21, 2, 7]. This is mostly because challenges like the n2c2 2006: De-identification and Smoking Challenge and the n2c2 2014: De-identification and Heart Disease Risk Factors Challenge, which were held to motivate research in healthcare textual data de-identification, provided guidelines and datasets which facilitated the advancement of research in this domain. We would like to expand further and tackle a general solution that can be adapted to any business domain, especially now that corporations and data holders from all different business fields are forced to respect privacy preservation

whenever applying data analytics, so they are in dire need of a domain-adaptive privacy solution.

In addition, the most prevailing method for privacy application is differential privacy [30, 31, 13], which aims at providing data anonymity by introducing noise to the existing data. However, this results in noisy outputs, which decreasing the accuracy of the analysis process, when one wants to run analysis on a differentially private dataset. These challenges motivate to initiate this research project, as the problem lies in the heart of real-world scenarios, with opportunities for a lot of problems to be fixed.

1.2 Author Verification

1.2.1 Background

Author verification (AV) is the field of work that specialises in answering the question whether a piece of text was published by a specific author or not [94]. Its utilities cover multiple areas of digital forensic text analysis like plagiarism detection [19] and fake news detection [88, 89]. Determining the true author of a document has been a task of social interest from the beginning of authorship attribution to words. Questions about the authorship of a document are of interest not only to field specialists (forensics and linguistics specialists, etc.), but also, in a much broader sense, for individuals from other fields of profession like politicians, journalists and lawyers. With the development of statistical techniques and the advances in text analytics, and because of the wide availability of accessible data online, the automated process of authorship analysis has become a very practical option, and the research interests in the field have greatly risen.

There are many problematic scenarios where which authorship analysis becomes a necessary component of solving them. Suppose a suspicious or malicious email is sent using an email account belonging to someone, who are subsequently accused of being the sender. Here there would be an investigation to verify if the accused, who is the real owner of the email address, is indeed the author or not. Or consider the scenario where online messages (like through social media, for example) or information are being traded under the name of a celebrity/influencer. Is that person actually spreading these information or not? This becomes an event of high importance because of the big effect that online influencers, like artists and giant social media users, have on the youth and growing minds, but also on the economical market in some cases [27].

AV frames the question of the true authorship of a piece of document as a classification problem: given an author A, a set of documents D authored by A, and a document d,

determine the possibility of whether d was authored by A or not. The vast majority of research about AV has been dedicated towards finding the authors of long texts. Long texts usually references documents of a substantial amount of information inside of them, like books, novels, and any body of text that extends over multiple full pages. In that regards, the task of AV is usually tackled by inferring linguistic characteristics (features) of the author by processing and analyzing documents written by them. These features allow us to create a model of the writing style of this author and measure how similar may any unknown document be to documents that are known to be written by that author.

However, with today's reliance on fast and short messages for communication, there is a need for AV on short texts more than ever. More specifically, there is a need for AV on micro-messages, like tweets. There is an important distinction to be made here, since processing short texts differs from that of long texts, mostly because of the lack of features to extract from short texts. Features that are deemed important by statistical models to infer an author's writing style. Previous work has shown that it is difficult to maintain good performance when an AV system that works on long text is applied to shorter text [63]. However, there has been many recent AV projects that experimented with short web data such as emails [1], web forum messages [92] and blogs [61]. Today, more and more interest has been assigned for AV on short texts. But, there remains a distinction between short texts like article excerpts and short anecdotes, and micro-messages which are the dominant means of communication on the web. Micro-messages range in length from a couple tens of words, down to 3 to 5 words per text. This comes with a great challenge, which is the lack of features to collect that would help to identify the author. Thus, AV is handled differently when working with long texts, short texts, and micro-messages.

AV methods can be generally divided into two categories: 1) similarity-based methods, in which authorship is verified by collecting and matching the author's writing style by some distance metric. 2) Machine learning methods, where machine learning models are used to learn specific features from the text that can uniquely identify the author. In both cases, the algorithms are extracting some stylistic features from the text, then finding a writing pattern to uniquely identify the author. These features include lexical features [81], character features [85, 51, 87], and syntactic features [1, 62]. These approaches have been applied for both long and short text, with varying degrees of success. Below is a list of all the aforementioned stylistic features.

Lexical features

- word length, sentence length, etc
- vocabulary richness
- word frequencies

- word n-grams
- spelling errors

Character features

- character types (letters, digits, punctuation)
- character n-grams
- compression methods

Syntactic features

- part-of-speech tags
- sentence and phrase structure
- grammatical and spelling mistakes

But a fundamental problem lies with such approaches, which is the heavy dependence on the textual structure. While this is not an issue when analyzing long texts, micro-message do not offer enough information per body of text to detect an author's stylistic pattern of writing. In addition, studies have shown that some stylistic features, like average word lengths, are neither stable within a single author, nor do they necessarily distinguish between authors [49]. In addition, an author's writing style can be easily spoofed by using an original message as a template, and then inserting/changing data in a way that alters the idea behind the message while preserving the writing style. From here, we aim to prove 3 hypotheses:

1. Text semantics can be used as a feature for AV.
2. Stylistic-based AV approaches don't take text semantics into consideration.
3. It is feasible to spoof a message in a way that fools style-based AV methods.

1.2.2 Motivation

The number of active social media users has reached 4.48 billion people as of January 2021 [37], which is up more than double from 2.07 billion in 2015. The distribution of these users can be seen in Figure 1.4, with Facebook still being the number 1 most used platform.

Although Facebook allows for $63,206$ characters per post, most posts shared are between 0 and 100 characters long, as depicted in Figure 1.3. So a lot of Facebook posts fall within the micro-messages domain. In addition, Twitter still holds 397 million users, and all posts shared on Twitter have a maximum posts count of 280 characters.

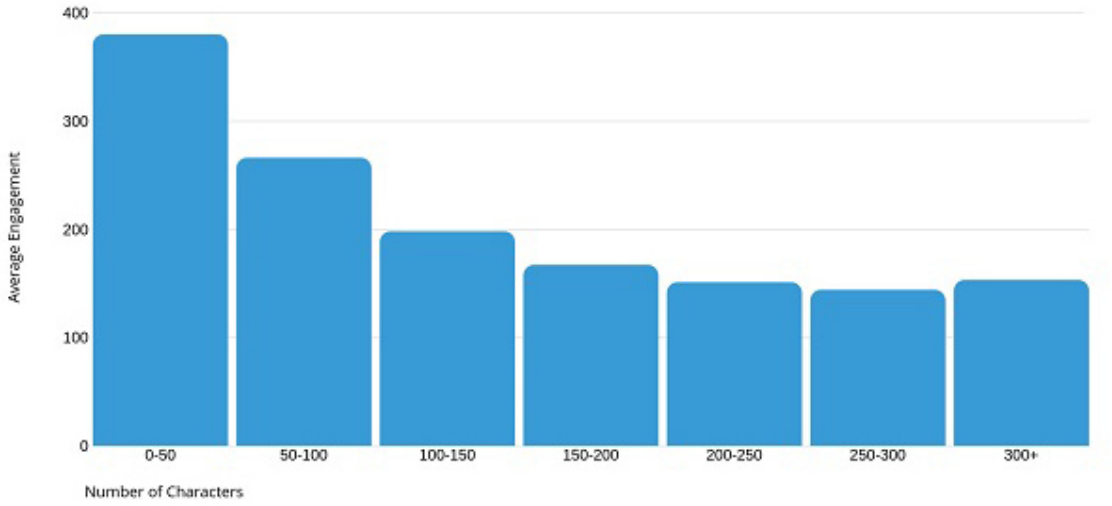

Figure 1.3: Facebook posts lengths in number of characters.
Source: Buzzsumo[2]

With this great growth of social media usage came a serge in the number of machine accounts that are designed to mimic human users [3, 23, 78, 20]. These accounts, known as Social bots accounts, have become more sophisticated and deceptive in their efforts to replicate the behaviors of normal accounts. Their behavior could either be harmless, like to promote material, or it could have malicious intentions like mimicking other users, or in some cases, account fraud [83]. These automated systems can produce high volumes of fraudulent information, so trying to detect them manually would prove fruitless, simply due to the sheer volume of interactions these bots are capable of.

The Challenges of Author Verification in Micro-Messages

Consider a VIP who's opinion can affect people's decisions, like a politician or an online influencer. This type of person would make a great target for imposters to try and imper-

[2]https://buzzsumo.com
[3]https://www.statista.com/statistics/617136/digital-population-worldwide/

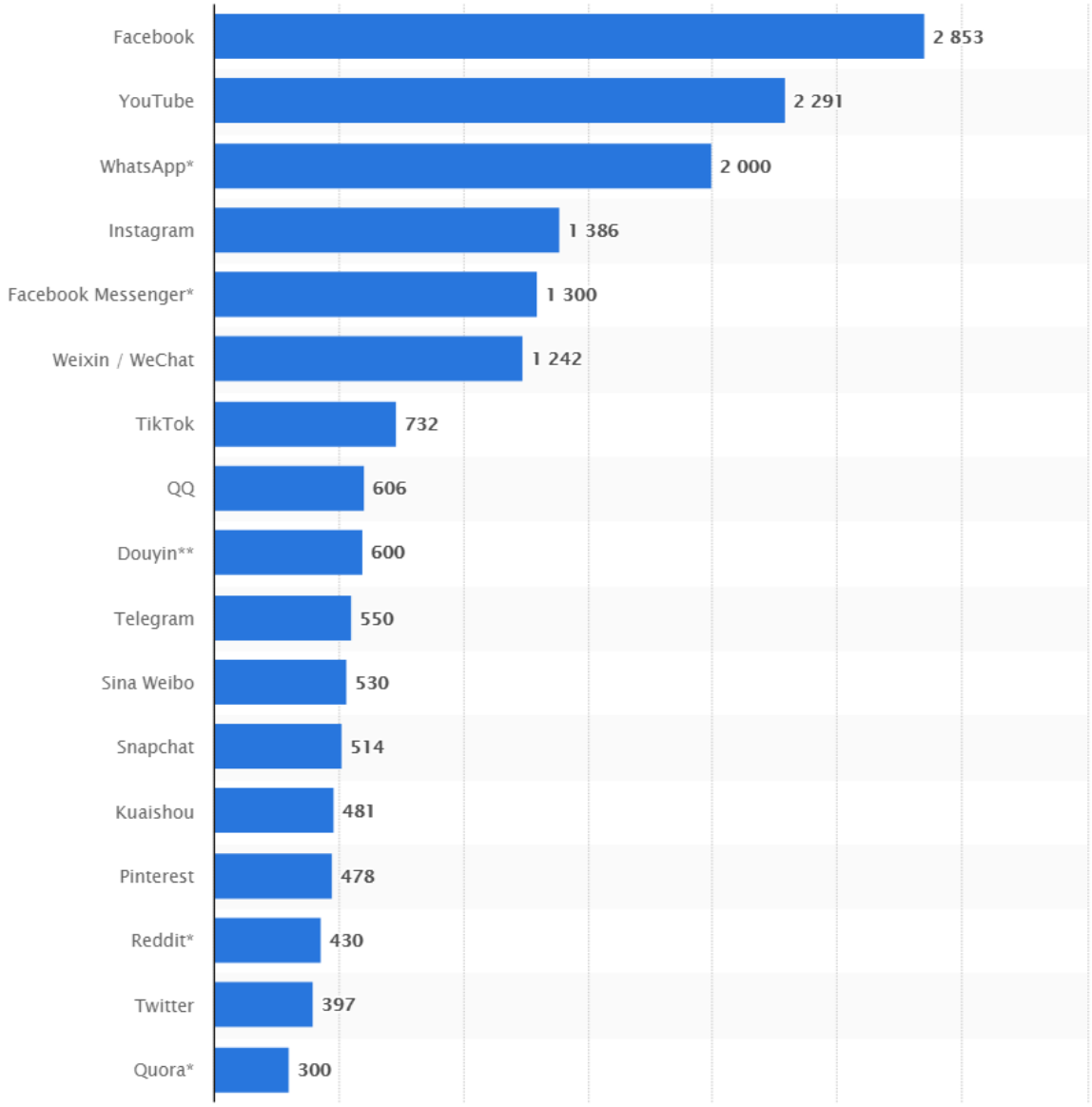

Figure 1.4: Number of social media users (in millions) as of July 2021.
Source: Statista Global digital population[3]

sonate with the aim of either spreading fake news, serving a particular agenda, or simply having a negative impact on said VIP's public image. A statement issued by this VIP can go as much as to affect the stock markets, like the rise in Signal's stock market value because of a tweet from Elon Musk recommending people to switch from using Whatsapp to using Signal [27]. Current style-based AV approaches suffer in performance when working with short text. However, when we consider the ability to imitate the author's way of writing, it might become impossible for those approaches to detect fake authorship. Consider the following quote from Elon Musk [102]:

*You **can't** have a person driving a two-ton death machine.*

If we simply remove 2 characters from the above quote, so that "can't" becomes "can", the same writing style would be maintained. An algorithm that is trained to detect Elon Musks's style of writing would still recognize the quote:

*You **can** have a person driving a two-ton death machine.*

as an Elon Musk quote. These kinds of edits would go by unnoticed by style-based AV systems since the doctored fake statement is fabricated following the structure of an authentic one, even though the meaning is completely changed.

We argue that for an AV system to work well with short text, it needs to go beyond style-based features and focus more on semantics. A good AV system should be able to not only detect an author‘s style of writing, but also have an understanding of their general opinions towards specific topics.

1.3 Contribution

We categorize our contributions in this thesis into primary and secondary contribution, presented below.

Primary Contributions

In this thesis, we focus on providing solutions to improve user privacy in textual data, and AV in micro-messages, specifically in the use case of social media. Our contributions are organized as follows:

1. **Anonymization of Personal Information in Textual Data**
   We provide a theoretical solution for privacy-preserving text analytics, detailed in

Chapter 3. We introduce the concepts of Identifiability, Sensitivity and Semantic Value, which are metrics calculated for each term (word) in a piece of text, which are then used to anonymize text in an adaptive manner based on the criticality of the business domain.

2. **Semantic-Based Author Verification in Micro-Messages**
   We introduce a novel approach for AV, detailed in Chapters 4 and 5. We base our approach on text semantics, and show that existing style-based approaches perform poorly when applied on short text. We provide static and dynamic parameter tuning methods using different statistical models. Dynamic parameter tuning is data-driven, meaning that the parameters are tuned from the text data itself.

3. **Impersonation Attacks on Existing Author Verification Models.**
   In Chapter 5, we show that stylistic-based approaches don't take semantics into consideration, and are thus vulnerable to impersonation attacks. We also show that our semantics-based approach significantly out-performs the state of the art in detecting impersonated texts.

4. **Temporal Effect of Data on Author Verification**
   In Chapter 6, we study the temporal effect of data on authorship verification, which is the fact that people change their opinions over time. We first synthesize a sample dataset of varying opinions over time, and use it to integrate the temporal aspect into the approach developed in Chapters 4 and 5. We then apply our findings on our influencers dataset (explained in **Secondary Contributions**).

Secondary Contributions

In addition to the main contributions discussed above, we've also introduced a new dataset and we did comparative experimentation with existing state of the art solutions of AV on micro-messages. The studies are briefly explained below:

1. **Literature Survey**
   A rigorous survey of the literature was conducted in order to find the shortcomings regarding privacy in textual data, and the inability for AV solutions to handle micro-messages. The survey covered all classical and advanced approaches from ones designed to handle structured to unstructured data for the privacy aspect, and the major competitions and the rule-based and machine-learning-based approaches for the AV aspect. Chapter 2 presents the results of the survey.

2. **A New Dataset of Twitter Influencers**
   For the purposes of our work, we collected a dataset composed of the tweets of the 86 most followed Twitter users. We started with the top 100 most followed user, but ended up with 86 after removing non-English users, and official news accounts. For each user, we collected a maximum of 3200 tweets.

3. **Testing the State of the Art in Author Verification on Micro-Messages**
   We've tested 2 of the top-performing algorithms in AV designed for long text on micro-messages: a stylistic-based method, and a deep-learning method. This is done for comparative purposes, and highlights the significance of using semantics in AV over stylometry when working with micro-messages.

1.4 Thesis outline

The remainder of this thesis is outlined as follows. In Chapter 2, we review the state-of-the-art methods for the problems of data privacy in text, and AV, which we research in this thesis. In Chapter 3, we present a theoretical approach for data privacy in text, which works on detecting identifiers and sensitive information in text, and applying the proper anonymization technique. In Chapter 4, we present a novel semantic-based AV approach, and address the deficiencies of style-based approaches for AV in micro-messages. In Chapter 5, we improve on the approach from Chapter 4, and we introduce a new dataset of Twitter influencers. We also perform comparative studies against the state of the art in AV, and show its inability to detect impersonation attacks. In Chapter 6, we study the temporal effect of data on AV. Finally, we conclude in Chapter 7, and discuss open research directions.

Chapter 2

Related Work and Background Information

2.1 Data Privacy and Anonymity

There are two natural models for privacy mechanisms: **interactive** and **non-interactive**. In the non-interactive setting the data collector, a trusted entity, publishes a "sanitized" version of the collected data; the literature uses terms such as "anonymization" and "de-identification". Traditionally, sanitization employs techniques such as data perturbation and sub-sampling, as well as removing well-known identifiers such as names and birth

dates, and social security numbers. Structured data were typically anonymized by simply removing all the explicit identifiers like names and phone numbers. However, in most of these cases, the remaining data can be used to re-identify individuals by linking it to other purposely collected data or by looking at unique characteristics in the released data [99, 6, 77, 6]. A more recent work by Narayanan et al. [77] shows a similar context, only this time de-anonymizing the Netflix Prize dataset users using publicly available Amazon review data [48, 74]. Here, they were able to uncover more user information like a user's full name and shopping habits.

2.1.1 Non-interactive Approach

k-Anonymity. k-anonymity [84] is a property of a dataset that describes its level of anonymity. Developed in 1998 as a means to address the problem of releasing person-specific data while preserving the anonymity of the individuals to whom the data refers using generalization and suppression techniques. A dataset is k-anonymous if every combination of identity-revealing characteristics (quazi-identifiers) occurs in at least k different rows of the dataset. Table 2.1 shows a dataset that has been 2-anonymized; note how the attributes "Age" and "Gender" are identical in the top 2 and bottom2 rows.

Age	Gender	Score
$[10-12]$	Male	98
$[10-12]$	Male	77
$[11-12]$	Female	97
$[11-12]$	Female	80

Table 2.1: 2-Anonymous Dataset

ℓ-Diversity. ℓ-diversity [71] was developed in 2006 to solve 2 privacy problems found in k-anonymity. First one is that an attacker can discover the values of sensitive attributes in a k-anonymous dataset when there is little diversity in those sensitive attributes. Second is background knowledge attacks. To give an example, if there are 100 different men with ages above 70 years living in area A who all have allergies to peanuts, then I know that Bob, who is 72 years of age, living in area A, also has an allergy to peanuts. ℓ-diversity aims to solve these problems by applying the following principle: a generalized quasi-identifier q$*$-block (equivalence class) is ℓ-diverse if it contains a minimum of 'ℓ' properly depicted values under the sensitive attribute present in these blocks. If every q$*$-block in a dataset is ℓ-diverse, then the dataset meets the ℓ-diversity concept. Table 2.2 shows an example of an ℓ-diverse (3-diverse) dataset.

t-Closeness. t-closeness [69] comes as a betterment of ℓ-diversity by decreasing the granularity of the interpreted data. Introduced in 2007, where Li et al. [69] showed that ℓ-diversity is neither necessary nor sufficient to prevent attribute disclosure, and instead provided t-closeness which requires that the distribution of a sensitive attribute in any equivalence class is close to the distribution of a sensitive attribute in the overall table. The distance between distributions is measured using Earth Mover's Distance (EMD). For a categorical attribute, EMD is used to measure the distance between the values in it according to the minimum level of generalization of these values in the domain hierarchy. Table 2.3 shows an example of a dataset that has 0.167-closeness with respect to Salary and 0.278-closeness with respect to Disease.

Nonsensitive			Sensitive
Zip Code	Age	Nationality	Condition
1305	≤ 40	*	Heart Disease
1305	≤ 40	*	Viral Infection
1305	≤ 40	*	Cancer
1305	≤ 40	*	Cancer
1485	> 40	*	Cancer
1485	> 40	*	Heart Disease
1485	> 40	*	Viral Infection
1485	> 40	*	Viral Infection

Table 2.2: 3-Diverse Dataset

ZIP Code	Age	Salary	Disease
4767*	≤ 40	3K	gastric ulcer
4767*	≤ 40	5K	stomach cancer
4767*	≤ 40	9K	pneumonia
4790*	≤ 40	6K	gastritis
4790*	≤ 40	11K	flu
4790*	≤ 40	8K	bronchitis

Table 2.3: Dataset with 0.167-closeness with respect to Salary and 0.278-closeness with respect to Disease

These methods are not applicable for providing privacy for unstructured data. The main focus of k-Anonymity, and by extension, the later improvements, is the anonymization of tabulated data. They were made at a time were structured data was the governing method for data storage. However, with the rise of unstructured data as the dominant form of

information storage, and textual data in particular, with is the dominant form of information transfer on social media, there came a need for a new method for privacy preservation.

2.1.2 Interactive Approach (Differential Privacy)

Differential privacy was introduced in 2006 by Dwork and Aaron [26]. It offers a robust mathematical definition of privacy, and was developed as a solution for privacy-preserving data analysis. It ensures that the result of an algorithm is not overly dependent on any instance, and states that there should be a strong probability of producing the same output even if an instance was added to or removed from the dataset. It is now used by the likes of Apple [4] and Google [28].

Although differential privacy is praised for being an interactive solution that can be adapted to different scenarios (data collection, data analysis, machine learning...), it is not without its flaws. Kifer and Machanavajjhala [58] show that it is necessary to make assumptions about how the data is generated, to provide privacy, which is unlike what differential privacy claims. In addition, the main criticism against differential privacy is the fact that it produces noisy results, decreasing the accuracy of the output.

2.1.3 Privacy in Textual Data

There has been lots of work on applying privacy to text, mostly in the form of de-identification. Challenges like the n2c2 2006: De-identification and Smoking Challenge [22] and the n2c2 2014: De-identification and Heart Disease Risk Factors Challenge [21] motivated research in textual data de-identification, namely in the field of healthcare. Performing such a task manually proved to be time-consuming and quite expensive. Douglass et al. [25, 24] reported that annotators were paid 50 US dollars per hour and read 20 000 words per hour at best. This motivated research in this domain to automate this process.

Earlier research in the field were oriented towards **rule-based** or **pattern-matching** solutions, using either complex regular expressions, dictionaries or a combination of both [8, 9, 33, 44]. The advantages of rule-based and pattern matching de-identification methods is that they require little or no annotated training data, and can be easily and quickly modified to improve performance by adding rules, dictionary terms, or regular expressions. The disadvantages are that developers have to craft many complex algorithms in order to account for different categories of PHI, and the required customization to fit a particular dataset. As such, PHI pattern recognition performance may not be generalizable to different datasets (i.e. data from a different institution or a different type of medical report).

Later work tended to be mostly based on machine learning methods to classify words as PHI or not PHI, and in different classes of PHI in the former case. The methods used a range of techniques from Support Vector Machines, to Conditional Random Fields, Decision Trees, and Maximum Entropy [5, 43, 100, 107]. More recent work is more focused on utilizing neural networks and deep learning in its approach to de-identify patient data. Ji Young Lee et al. [68] incorporate human-engineered features as well as features derived from electronic health records to a neural-network-based de-identification system composed of a Long Short Term Memory neural network [86].

2.1.4 Differential Privacy with Textual Data

Benjamin Weggenmann et al. provide an automated text anonymization approach that applies differential privacy to the vector space model [106]. They obscure term frequencies in textual documents' TF-IDF vectors in a differentially private manner. Their aim is to prevent a document's author attribution through the evaluation of the document's TF-IDF vectors using different data-mining techniques. They also demonstrate that this approach has a low impact on accuracy when mining these document vectors. Our goal is different from that of Weggenmann in that we aim to provide privacy methods to the actual text documents, and not their vector representations.

2.1.5 Discussion

The application of **differential privacy** to textual data is mostly possible by using the word vectorization of key terms chosen from the text, then applying differential privacy to these vectors. This is useful for running privacy-preserving statistical analysis on these terms, or to prevent a document's author attribution [106]. However, it falls short when it comes to preserving the structure of the text, since it picks out only specific terms discarding the rest of the text. Our aim is not to choose specific keywords from a given text to run a specific analysis, rather we want to conserve most of the text, removing/obscuring only what's necessary to preserve the privacy of any individuals present in it while preserving most of the text's utility.

Dictionary based and **pattern-matching based** approaches to provide privacy in text, like MITdeid [56] and NLM-Scrubber [57], aren't much complex to implement, and require little to no annotated training data. But that comes on the expense of being static, where every target term to be captured has to be manually transcribed through complex regular expressions. In addition, solutions using these approaches can't be generalized to handle different datasets; they're only made to handle a target dataset, or a category of datasets.

Neural-network based approaches like DeID and CliniDeID [59] are the ones with the most promise in terms of accuracy, adaptability, and generalizability. Once a neural network model is created and trained to capture specific terms in a piece of text, it is then able to capture other terms that are symantically equivalent, which are learned from the context of the text. This is crucial in the field of text analysis since it is very difficult to predict every possible structure of a sentence in a given language, or every possible use of a term or word. The problem with available solutions is that they are fixated on the field of medicine and capturing PHIs, and don't take info account texts about other domains like finance or trading. Besides, these solutions treat all identifiers equally, although one identifier (like names) can have higher priority to be removed than others (like age) since it can identify an individual more easily.

2.2 Author Verification

Authorship analysis has been studied extensively in the context of the *author identification* problem, in which the author of a questioned document is to be selected from a small set of candidate authors based on some metrics related to the manner the document is written in. In the scope of this work, it is necessary to make the distinction between *author verification*, the focus of this work, and the closely related *author attribution*. Authorship verification is a fundamental problem in authorship attribution since any problem can be decomposed into a set of verification problems. It is quite challenging since a verification model should estimate whether the disputed text is similar enough to the given texts by a certain author while an attribution model should estimate who the most similar candidate author is.

2.2.1 Authorship Attribution

Broadly explaining, *authorship attribution* is the attempt to infer the characteristics of the creator of a piece of linguistic data. Our work focuses on textual data, but authorship attribution can be applied to any type of linguistic data, like audio speeches. There are mainly three problems in authorship attribution.

- **The "closed class" problem:** given a particular sample of text known to be by one of a set of authors, determine which one.
- **The "open class" problem:** given a particular sample of text believed to be by one of a set of authors, determine which one, if any, is the author.
- **The "profiling" problem** determine any of the properties of the author(s) of a sam-

ple of text. For example, was the document written by one or multiple authors? was the document written in the native tongue of the author? What's the author's sex?

2.2.2 Authorship Verification

Author verification, on the other hand, is the task of deciding whether two texts or documents were written by the same author, by comparing the writing styles of the 2 documents. This is a broad definition that can be extended to suit more complex or more specific scenarios. In our case, we are working on social media posts, which do not provide much information to consider comparing only 2 documents to verify same authorship. Thus, we use the following problem definition: given an author A and a set of documents D known to be published by A, and a document d_x of questioned authorship, determine whether d_x was published by A or not. This verification process is demonstrated in Figure 2.1.

Figure 2.1: The authorship verification problem.

2.2.3 The PAN at CLEF Tasks

A lot of works have been presented for AV in long texts. Most prominent are the PAN at CLEF challenge series[1], which has been established as one of the main forums of text mining research focusing on the identification of personal traits of authors left behind in texts unintentionally. It is considered one of the most important benchmarks and references for authorship attribution research. The PAN authorship verification tasks [55, 95, 96, 54] tackle what Kopel et al. [64] called the "fundamental problem" in authorship attribution: Given two documents, are they written by the same author? The PAN series started in 2011 as a yearly challenge to solve research problems related to: Authorship Analysis,

[1]https://pan.webis.de/shared-tasks.html

Textual Feature Category	Textual Features
Character	letter frequencies, punctuation mark frequencies, character n-grams, and common prefixes-suffices of words.
Lexical	word frequencies, word n-grams, function words, function word n-grams, hapax legomena, morphological information (lemma, stem, case, mood, etc.), word / sentence / paragraph length, grammatical errors and slang words.
Syntactic	POS tag counts, POS n-grams

Table 2.4: Most popular textual features collected by submissions to the PAN 2013, 2014, and 2015 AV challenges.

Computational Ethics, and Originality. In the domain of Authorship Analysis, they tackle the problems of: Authorship Attribution, Authorship Clustering, Authorship Verification, Author Masking, and Author Profiling amongst others. In the PAN Authorship Verification challenge of 2013[55], 2014[95], 2015[96], and 2020 [54], the task was focused on AV in long texts. They covered multiple languages, including English, Dutch, Greek and Spanish. The participants' submissions relied mostly on stylistic analysis of the text documents to infer authorship, with heavier focus on machine learning models in the PAN 2015 and 2020 competition. The most popular textual features collected by participants' solutions are presented in Table 2.4.

The data provided for analysis is considered long text. Table 2.5 presents all the datasets' average word count for every language in the PAN 2013, 2014, 2015 and 2020 AV tasks. We can see that the average word count per document exceeds 1000 words in most datasets for the PAN 2013, 2014 and 2020 AV tasks, and is regularly above 350 words in the PAN 2015 AV task.

Criticizing PAN at CLEF

Bevendorff et al. [10] review the PAN authorship verification task and state that the experiment design presented at PAN may not yield progression of the state of the art. They tested what they call a "Basic and Fairly Flawed" authorship verifier model which performs competitively with the best approaches submitted to PAN until that time, which were the PAN 2013, 2014 and 2015 AV tasks [55, 95, 96].

PAN 2013			
	Language	**Documents**	**Avg. words documents**
training	English	44	1159
	Greek	129	1484
	Spanish	19	723
testing	English	155	1204
	Greek	179	1445
	Spanish	90	961
PAN 2014			
	Language	**Documents**	**Avg. words documents**
training	English	929	1340.9
	Dutch	470	283.4
	Greek	385	1404
	Spanish	600	1135.6
testing	English	1118	2718.9
	Dutch	489	281.6
	Greek	368	1536.6
	Spanish	600	1121.4
PAN 2015			
	Language	**Documents**	**Avg. words documents**
training	English	200	366
	Dutch	276	354
	Greek	393	678
	Spanish	500	954
testing	English	1000	536
	Dutch	452	360
	Greek	380	756
	Spanish	500	946
PAN 2020			
	Language	**Documents**	**Avg. words documents**
training	English	552000	4200
testing	English	33800	4200

Table 2.5: Overview of the PAN 2013, 2014, 2015, and 2020 authorship verification corpus

2.2.4 Other AV Approaches

Outside the PAN competition, there has been lots of works presented for AV in long texts [14, 16] and short texts [11], and there has been many recent AV projects that experimented with short web data such as emails [1], web forum messages [92] and blogs [61]. But to the best of our knowledge, not much work aims to tackle AV on micro-messages.

Burgebacher et al. [15] introduce an interesting user verification system based on gesture typing, which analyses the user's keyboard typing pattern in real life to verify a user. Suman et al. [98] introduce a multi-modal Siamese-based framework for AV to extract features from texts, and they put emphasis on an author's preferred usage of emojis as an additional feature. Boenninghoff et al. [11] propose ADHOMINEM, a new attention-based neural network topology for similarity learning in short text. The network applies characters-to-word, words-to-sentence and sentences-to-document encoding to extract document-specific features to make a similarity analysis between two documents. They apply their model to a dataset of amazon reviews which they develop themselves for this purpose. Buddha et al. [14] provide a new approach for authorship verification using a non-uniform distributed term weight measure to calculate term weights in the text instead of TF-IDF. Castro et al. [16] propose a solution for AV based on comparing the average similarity of an unknown authorship text with the Average Group Similarity between text samples from the target author. They also performed experiments with a total of 17 types of linguistic features, grouped int 3 categories: character, word and syntactic, and used six similarity functions. Halvani et al. [45] propose an AV approach that considers only topic-agnostic features in its classification decision.

Van Dam et al. [19] investigate the influence of topic and time on AV accuracy. Regarding topic influence, they found that cases with documents of similar topics overall (positive and negative) were found to increase accuracy of AV. As for the influence of time, they found that writing style indeed changes over time, by comparing Wikipedia Talkpages contributions made within a week with Wikipedia Talkpages contributions made years apart. AV is more accurate when comparing texts that have been written within a short period of time. Schwartz et al. [85] did work on micro-messages, but that was for the task of authorship attribution, and not authorship verification.

2.2.5 Performance Measures

The standard for evaluating the performance of AV models is done through the metrics: precision, recall, F-1 score, area under the curve (AUC), correctness at 1 (c@1) [80] and accuracy. Precision quantifies the number of positive class predictions that actually belong

to the positive class. Recall quantifies the number of positive class predictions made out of all positive examples in the dataset. F-1 score provides a single score that balances both the concerns of precision and recall in one number. c@1 is an extension of the accuracy measure that reward systems that maintain the same number of correct answers and at the same time decrease the number of incorrect ones. The accuracy rate represents the percentage of correctly classified observations from both classes.

To evaluate these metrics, we use the following concepts:

- **True Positive (TP):** the model correctly predicts that A is indeed the author of d_x.
- **True Negative (TN):** the model correctly predicts that A is not the author of d_x.
- **False Positive (FP):** the model incorrectly predicts that A is the author of d_x, while indeed A is not.
- **False Negative (FN):** the model incorrectly predicts that A is not the author of d_x, while indeed A is.

Table 2.6 shows the confusion matrix of the aforementioned concepts.

Predicted	**Actual**	
	True Author	**Fake Author**
True Author	True Positive (TP)	False Positive (FP)
Fake Author	False Negative (FP)	True Negative (TN)

Table 2.6: Confusion matrix form

It is worth noting that each of these performance measures cannot be used alone to confirm the competitive quality of the methods.

Chapter 3

Anonymization of Personal Information in Textual Data: a Theoretical Approach

3.1 Preliminary Definitions: Identifiability and Sensitivity

In this work, we are concentrating on text, therefore all operations are done on textual documents. Each of which contains natural language text. We assume that a document is associated with a single person and without loss of generality, we assume that multiple documents may refer to a single person.

In order for a document in question to cause a privacy problem, it should contain identifying information enough to uniquely identify its associated person, and should contain private or sensitive information. If any of these conditions are not satisfied, then the document will

not cause privacy leaks. Therefore, the degree of privacy risks associated with a document are a combination of the identifying information and the private information present in it. Our objective is to:

1. provide a metric to assess the degree of identifiability a given textual document. That is to say. to what extent can the person who's information are present in the document be re-identified.
2. provide a metric to assess the sensitivity of the private information contained in said document.
3. finally, provide a methodology to de-identify the person mentioned in the document so that the degree of risk of re-identification is below a certain threshold.

This threshold depends on the sensitivity of the private information present in the document, which is dependent on the field of work in which the document exists. For example, a document containing a person's medical history would be ranked higher in terms of sensitivity to one containing a purchase history, hence the threshold should be lower (more strict) for the medical document. We refer to the sensitivity of the information inside a document as the sensitivity of a document, which indicates how risky it is to publish this document as is from the perspective of privacy infringement. We refer to words in documents as "terms", but not all words, rather ones that convey meaning, and are not "stop words".

We characterize a given document based on its terms as follows: each term in every document has 2 attributes. For each attribute, we can provide a metric value indicating the significance of the term with respect to said attribute. Each of the 2 metrics can be considered as a normalized weight, where a higher value indicates that the term is more critical. These attributes are: **identifiability** and **sensitivity**. Each of the aforementioned attributes serves a purpose in determining whether a document needs to be anonymized or not, before either sharing with third parties or running data analysis on it. Sample term metrics are provided in table 3.1, and an overview of the document processing procedure is displayed in figure 3.1.

3.1.1 Identifiability

Identifiability is the degree to which a term can identify an individual. For example, an email address has high identifiability, since alone, it is capable of uniquely identifying a person. On the other hand, quazi-identifiers, like Age, Sex, and Location, have lower identifiability scores, since alone they may not be able to uniquely identify in individual.

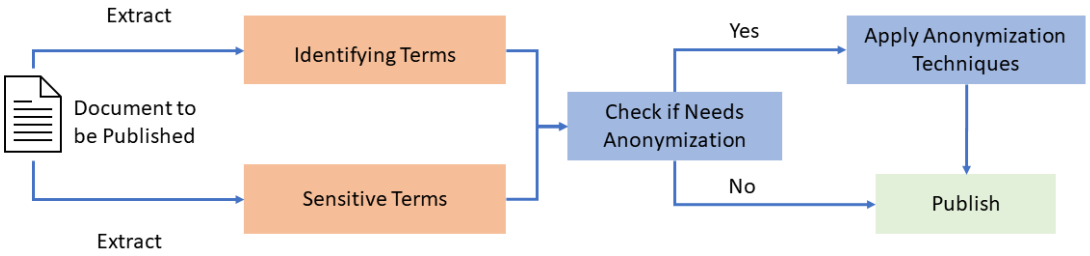

Figure 3.1: Document Processing Procedure.

Total identifiability of a document is calculated as the sum of the identifiability score of all of its terms, as shown in equation 3.1, where n is the number of terms in the document. If the final score is above a certain threshold (to be determined by the application user), then the document qualifies for anonymization procedures, provided that its sensitivity score (mentioned below) is also above the sensitivity threshold. The justification behind the use of this calculation is as follows: consider 2 documents A and B, and assume that A has 1000 terms, and B has 20 terms. Say that both A and B have only 10 terms of high identifiability. Even though A has a much lower respective count of identifying terms than B, those 2 documents are identical in their ability of identifying their respective individuals. It's enough to have one sentence in the beginning of a document stating that all information in said document reference a person p. Hence, we use the total sum of identifiability score of terms rather than, say, the average score.

$$Total\_Identifiability\_Score = \sum_{i=1}^{n} ident\_score(term_i) \tag{3.1}$$

3.1.2 Sensitivity

Sensitivity is an indication to whether or not a term reflects sensitive information. For example, the name of a disease that an individual might have. Terms with a high sensitivity score signify that the parent document is a sensitive one, and thus needs to be anonymized before sharing it, otherwise it risks leaking sensitive information. Each domain of business (finance, medicine...) has its own set of sensitive keywords, so this is domain-specific.

In a similar fashion to calculating a document's total identifiability, the total sensitivity of a document is calculated as the the sum of the sensitivity score of all of its terms, as shown in equation 3.2. If the final score is above a certain threshold (to be determined by

Term	Identifiability	Sensitivity
age	0.3	0.2
sex	0.4	0.3
aids	0.1	0.7

Table 3.1: Sample of term identifiability and sensitivity metrics

the application user), then the document qualifies for anonymization procedures, provided that its identifiability score is also above the identifiability threshold.

$$Total\_Sensitivity\_Score = \sum_{i=1}^{n} sens\_score(term_i) \tag{3.2}$$

3.2 Constructing the SEEDs

In this section, we will introduce the seed identifier keywords $SEED_I$ and the seed sensitive information $SEED_S$ which are domain-specific collections of terms that we use in the anonymization process.

The issue of how to measure the identifiability and sensitivity of a term is a challenging research problem which will be addressed in the context of this work. However, our privacy preservation method based on the 2 mentioned measures will not dependent on how these measures are obtained. Therefore, we can assume a list of $seed\ keywords$ containing the identifiers and sensitive words for each given document, with the above 2 metrics for each of the terms in the universal set of terms.

Due to the nature of text documents having different priorities to what is private and what is sensitive based on the *business domain* that said documents belong to (finance, healthcare...), it is important to take into account the differences of each term's "criticality" in a given document based on the business domain. For example, regarding sensitivity, a medicine name would be considered a sensitive term in the medical domain, since it will most likely be referenced to a certain patient. On the other hand, the same medicine name would not be considered as sensitive in the trades business domain, since a merchant could be simply buying this medicine without the need of personal use. Thus, individual research could be done in every business domain to extract the categories of critical information. As an example of a business domian defining its own identifiers, the Health Insurance Portability and Accountability Act (HIPAA) [97] placed regulations that list the identifying information, referred to as protected health information (PHI), that must be removed from documents

before publishing in order to preserve patient confidentiality. In the United States, HIPAA defines 18 different types of PHI:

1. Names
2. Dates, except year
3. Telephone numbers
4. Geographic data
5. FAX numbers
6. Social Security numbers
7. Email addresses
8. Medical record numbers
9. Account numbers
10. Health plan beneficiary numbers
11. Certificate/license numbers
12. Vehicle identifiers and serial numbers including license plates
13. Web URLs
14. Device identifiers and serial numbers
15. Internet protocol addresses
16. Full face photos and comparable images
17. Biometric identifiers (i.e. retinal scan, fingerprints)
18. Any unique identifying number or code

This list of PHI form what we call in our work the seed identifier keywords $SEED_I$. The same procedure can be done to list the sensitive terms in a business domain, and create the seed sensitive information $SEED_S$. Once these seeds are created, we can use these terms in the following analysis steps, that place a criteria on a document as being in need of anonymization or not.

3.2.1 Ranking SEED Terms

Ranking $SEED_I$ Terms

Consider a set of documents D belonging to a specific business domain. These documents contain information about a set of people P. Given $SEED_I$, the aim is to rank each element of $SEED_I$ by order of how well can it identify a person belonging to P. For example, a person's name might be sufficient to uniquely isolate them from the rest of the people in P, while their age might not be, since there might be multiple people in P with the same age. Hence the entity *Name* would be ranked higher in $SEED_I$ than *Age* in terms of identifiability. In addition, an existing $SEED_I$ can be built upon, like using the PHI from HIPAA (section 3.2) as a base seed, and adding/removing categories as necessary. This ranking process can be done once on a large-enough sub-sample of D, providing an identifiability metric weight to all categories in $SEED_I$, and then the terms of these seeds can be ranked by analyzing their use in the respective domain.

Ranking $SEED_S$ Terms

Ranking sensitive information is more reliant on the business domain that a document belongs to, and requires good knowledge of B as sensitive information are likely to be unique to B, more so than the identifiers. However, the same cannot be done with $SEED_S$ as the sensitive information in the medical domain, like diseases and medications, are not likely to be present in other business domains, like finance. Hence, constructing $SEED_S$ is to be done strictly on a per-domain basis, and in decreasing order of sensitivity. The rank of a term in $SEED_S$ would determine its metric, from most critical to least critical.

3.3 Framework For Anonymization

3.3.1 Term Extraction Using NER

Named Entity Recognition (NER) is a sub-problem of information extraction that can be used to identify particular entities, such as people, places, companies and organizations. NER is used in our work to extract the identifiers and sensitive information from text. Starting with a list of seed identifier keywords $SEED_I$ and seed sensitive information $SEED_S$, we can build a specific NER model to target a specific business domain.

There exists multiple tools that can perform NER, like spaCy [50], PolDeepNer [73], POS-BIOTM [93] and Stanford NER [32] to name a few. They all serve a similar purpose: to

extract specific features from text based on a predefined set of rules. These rules can be manually annotated (rule-based), or produced from training with machine learning and deep learning approaches, or a hybrid of both techniques. In [40, 108], the authors conclude that rule-based approaches seem to outperform the machine learning ones, with the additional costs of time and effort, in addition to the need of the experience of domain experts. However, in [109, 41], The authors present a contradictory result, where deep learning and hybrid models seem to outperform rule-based ones. In our work, we specify the need to develop/adapt an NER system per business domain. Each new development process needs to adapt to the identifiers and sensitive keywords associated with each respective domain. We make no preference between the use of a rule-based or a machine learning model. There are costs for both approaches, as a rule-based NER system is costly in terms of time and labor, while a machine learning model is costly in terms of its need of high quality labeled data, keeping in mind that there needs to be plenty of data for each new keyword in $SEED_I$ and $SEED_S$, which might not be a guarantee for every business domain. In addition, NER systems can be extended or adapted to suite new entities [93, 79, 82]. On this basis, we can use the pre-constructed $SEED_I$ and $SEED_S$ to extend an existing NER system capable of extracting identifiers and sensitive information from text documents.

3.3.2 The Anonymization Process

For a given business domain D, after the creation of $SEED_I$ and $SEED_S$, and an NER system is tuned to operate on them, we are now able to anonymize documents belonging to D. Consider a document d belonging to D, the anonymization process is done in 3 steps:

1. **Terms extraction.** Using the tuned NER system, sensitive and identifiable terms are extracted from d, and provided with their associated metric from $SEED_I$ and $SEED_S$.

2. **Criticality assessment.** The $Total\_Identifiability\_Score$ and $Total\_Sensitivity\_Score$ are calculated. If either of the $Total\_Identifiability\_Score$ or $Total\_Sensitivity\_Score$ is below the associated identifiability / sensitivity threshold, then no further action is needed, as d either does not have enough identifiers to reveal the identity of the person mentioned inside of it, or d does not contain enough personal or sensitive information to be considered a sensitive document. Otherwise, the final anonymization step is needed.

3. **Identifiers masking.** For all identifier terms extracted from d, apply a masking technique where each term is replaced by its associated categorical term from $SEED_I$. Example, a person's first name would be replaced by the categorical item $First\_Name$.

Chapter 4

Semantic-Based Author Verification in Micro-Messages

In this Chapter, we focus on the problem of Author Verification (AV) in social media. More specifically, we experiment with AV on twitter tweets, although our approach can be applied to any form of short text. We experiment with one of the top performing algorithms of the PAN 15 AV task, which caters for one of the most important benchmarks to which new AV approaches refer and compare against, and show that it performs poorly when handling micro-messages, and we propose a novel approach for AV, which is a sentiment based author verification method for short text. Also, we experiment with our sentiment-based author verifier on a tweets dataset, and analyze the results and performance.

4.1 Preliminary Information

4.1.1 Sentiment Analysis

Sentiment analysis, also known as opinion mining, is a natural language processing technique used to determine whether a text is positive, negative or neutral. It is used to study people's opinions towards entities such as products, services, individuals, issues, events, topics, and their attributes [70]. This field of study saw a rapid growth with the growth of social media on the Web (reviews, discussion forums, Twitter, ...) because of the huge volume of opinionated data recorded in digital form. A significant goal of sentiment analysis is to classify and analyze the reviews related to products, hotels, online booking sites, e-commerce sites, social media, etc [46, 91]. In our work, we follow a similar strategy, where we analyze the opinions of an author A relative to specific topics or keyphrases, and use that to deduce whether or not a document of questioned authorship d_x could have been authored by A or not.

Different Techniques of Sentiment Analysis

Sentiment analysis relies on two types of techniques, lexicon based and machine learning based techniques [90]:

- Lexicon based or corpus based techniques: These techniques are based on decision trees such as Sequential Minimal Optimization, Hidden Markov Model, k-Nearest Neighbors, Single Dimensional Classification, and Conditional Random Field, related to methodologies of sentiment classification.
- Machine learning based techniques: mainly come in 2 flavours: unsupervised and supervised. **Unsupervised learning** conduct clustering, that it clusters texts together in terms of similarity with respect to a set of engineered features, designed to reflect a text's sentiment. **Supervised learning** is based on labeled datasets, and thus the labels are provided to the model during the learning process, and the model then learns to associate the labels provided to a set of engineered features to be extracted from the text.

Sentiment Value Interpretation

Generally, sentiment analysis is used in categorizing text into 3 categories of sentiment, *po-*

sitive, *negative* and *neutral*. There are also the case of fine-grained sentiment analysis, where there exists 5 categories of sentiment, *negative*, *slightly negative*, *neutral*, *slightly positive* and *positive* [65, 18]. In our case, we are more interested in sentiment as a continuous value represented as a real number between -1 and 1, where -1 represents the most negative sentiment, 1 represents the most positive sentiment, and generally, a value between -0.5 and 0.5 represents a neutral sentiment. Many existing models and tools are able to produce such representations of sentiment, but in a lot of scenarios, the values are actually rounded into the aforementioned sentiment categories.

4.1.2 Word Embedding

Word embedding designates a set of machine learning techniques which aim to represent the words or the sentences of a text by vectors of real numbers, described in a vector model (or Vector Space Model). These new representations of textual data have improved the performance of automatic language processing methods, such as Topic Modeling.

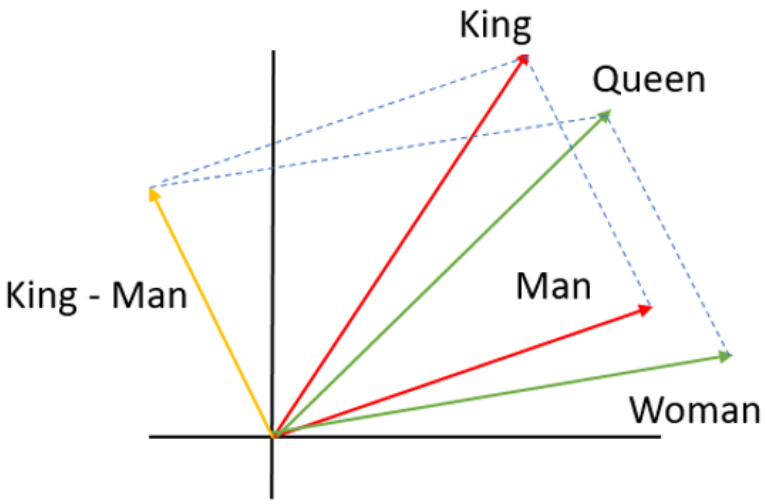

Figure 4.1: Word vector operations representation

Word embedding is based on the linguistic theory founded by Zelling Harris [47] and known as Distributional Semantics. This theory considers that a word is characterized by its context, that is to say by the words which surround it. Thus, words that share similar contexts also share similar meanings. The algorithms of embedding word is most often used to describe words through digital delivery, but can also be used to construct vector representations of whole sentences, biological data such as DNA sequence, or of the represented networks like graphs.

There are several approaches to word embedding. The first date back to the 1960s and are based on dimensionality reduction methods. More recently, new techniques based on probabilistic models and neural networks, such as Word2Vec [75], have made it possible to obtain better performance. As words are represented in the vector space, one can apply vector operations on them, and get meaningful results. Probably the most famous example of such operation is taking the vectors for King, Man and Woman, you can calculate King - Man + Woman and then you'll get the vector for: Queen, as presented in Figure 4.1.

Dimensionality Reduction

Dimensionality reduction denotes any method that makes it possible to project data from a large-dimensional space into a smaller-dimensional space. It helps reduce the complexity of a machine learning problem on several levels:

- from a theoretical point of view, this automatically leads to an improvement in the stability and robustness properties of the algorithms [12]
- from a practical point of view, this simplifies the resolution of the associated optimization problem, by reducing the space of solutions. In other words, reducing the dimensionality limits the number of possibilities to test, which allows the data to be processed faster.

This operation is crucial in visualizing word vectors, since we are unable to visualize beyond 3 dimensions. The word "dimension" is used here in the algebraic sense, ie the dimension of the vector space underlying the values of the vectors of descriptors. Among the most well-known methods for dimensionality reduction are:

- PCA (Principal Component Analysis) and its variants, which consist in identifying the main directions (combinations of descriptors), ie those which concentrate the most variance [34].
- ICA (Independent Component Analysis) which also seeks to identify orthogonal directions, i.e. uncorrelated from each other [17].

4.2 System Framework

In this section, we detail our approach for sentiment-based author verification. The basic idea behind our work is to use the semantics of the text to infer whether or not a certain

author could have authored it or not. The state of the art techniques focus on text structure, like extracting grammatical mistakes and writing patters, which does not yield the best results in micro-messages, like Twitter tweets. Given an author A, and a document of questioned authorship d_x, our model handles author verification in 2 steps. Step 1 is to construct what we call the Opinion History of A, which gathers A's opinion about the different topics that A have mentioned before in any published text. Then, in Step 2, we check the sentiment of the topic t mentioned in d_x, and compare that to A's sentiment about t, from the Opinion History. If the sentiments are not within a certain threshold from each other (don't match in polarity), then it's unlikely that A is the author of d_x.

This is based on the idea that it is unlikely for a person to suddenly change their opinion drastically regarding a certain subject. For example, if someone is well known to like Apple phones, it is unlikely that one day they would start talking negatively about them. We can use this knowledge to create a system that checks for inconsistencies between the history of opinions belonging to the author A. There is of course the argument that people do change their opinions over time, and this is studied further in Chapter 6 of this work. This approach can be applied to short texts, long texts, social media posts, and even excerpts of quotes. As long as a piece of text is known to belong to A, it can be divided into individual sentences, and the associated keyphrase/opinion pair can be extracted and added to the opinion history we have about A.

A visualization of the overview of our model can be seen in Figure 4.2. For the rest of this section, assume the following terminology: d_x is a document of questioned authorship, claiming to be authored by an author A, and D is a set of documents known to have been authored by A. The approach is detailed hereafter.

4.2.1 Creating the Opinion History

In this section, we will explain the details of creating the opinion history of A. The Opinion History serves as a reference store for A's opinions regarding different topics. We start with a set of texts or documents (tweets) which are known to be published by A, and then extract the keyphrase kp and the associated sentiment s of each of these documents. The process of creating the Opinion History is depicted in Figure 4.3.

Keyphrase extraction and embedding.

A keyphrase is the main topic that a piece of text revolves around. For example, the keyphrase of the sentence *"I like to use Apple phones"* is *Apple phones*. Given an Author

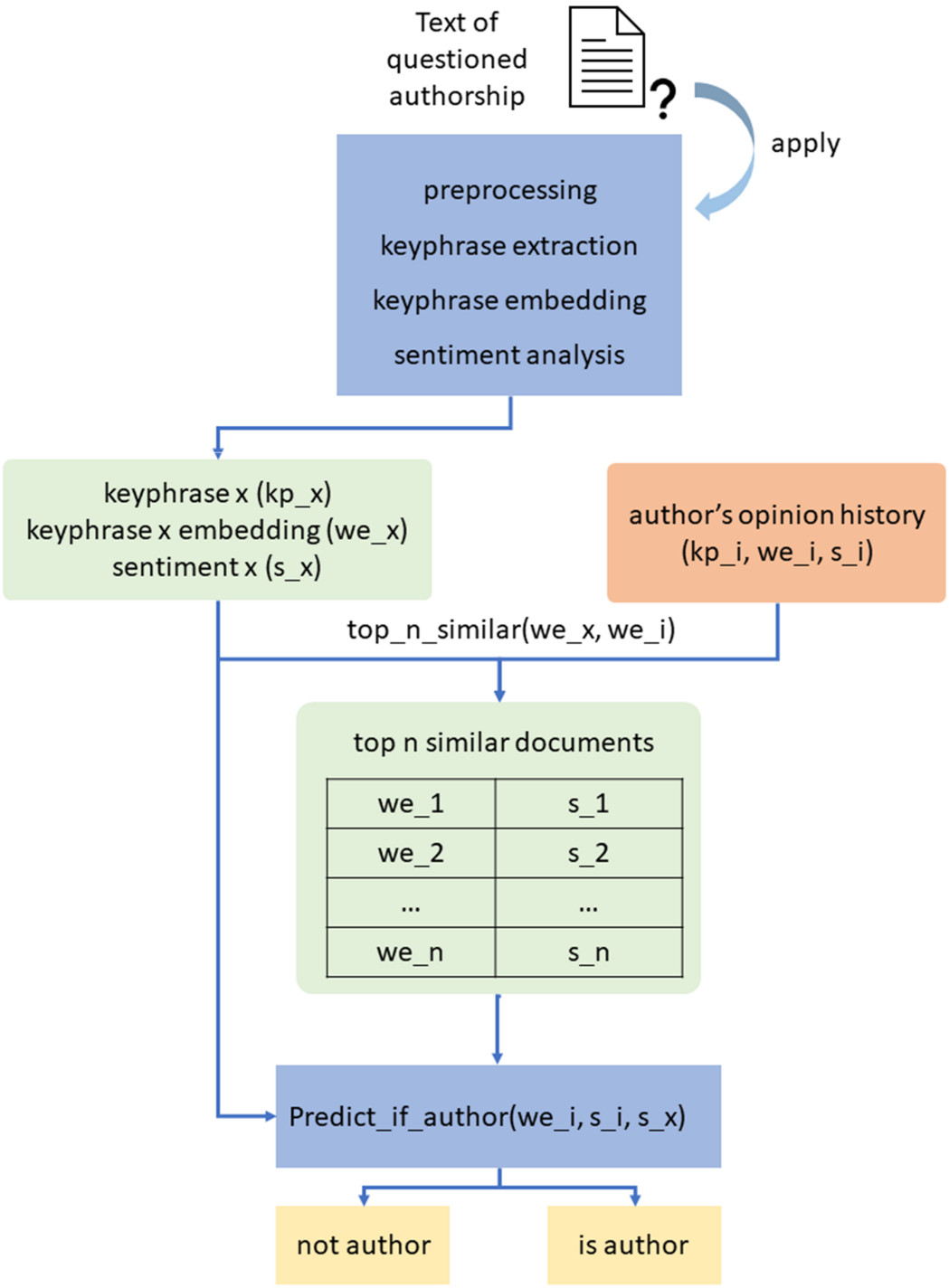

Figure 4.2: Sentiment-based author verifier architecture.

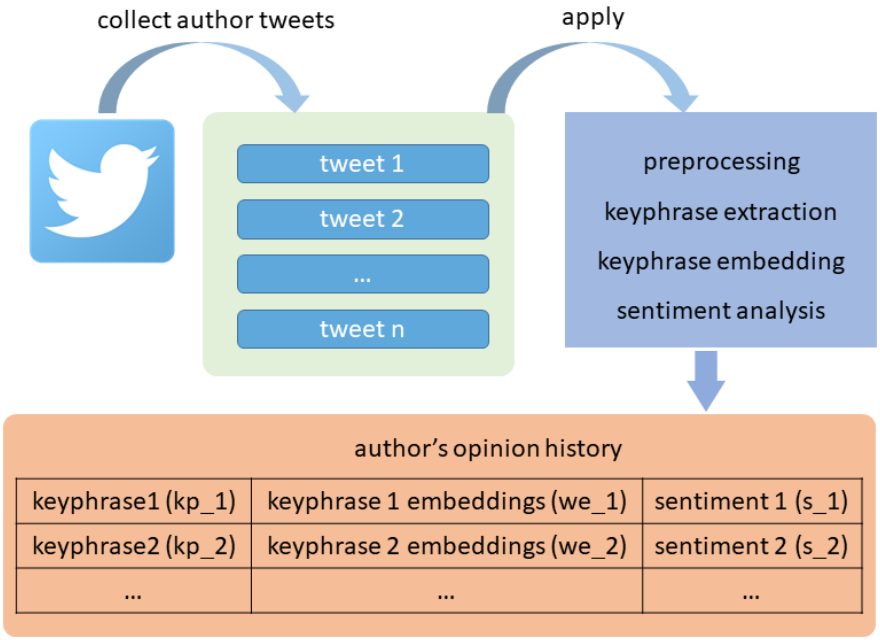

Figure 4.3: Creation of the Opinion History

A suspected of publishing a piece of text d_x, we collect text previously authored by A, and then extract the keyphrases. A keyphrase acts as the main topic that the text is centered around. For every extracted keyphrase kp_i, we calculate its word embeddings we_i. This allows us to do similarity search between keyphrases in a later stage. In the scenario where our text of unknown authorship centers around a keyphrase kp_x that has not been precisely encountered by our model before, we can estimate it by looking for the keyphrases most similar to kp_x. Word embeddings, combined with the sentiments extracted in step 2, form what we are calling the opinion history that is used for authorship inference. Algorithm 1 presents the opinion history creation process. By using word embeddings, running a similarity search algorithm, like cosine similarity, between the word embeddings of every keyphrase in the opinion history, and the word embeddings of kp_x returns the keyphrases that are most semantically similar to kp_x. The similarity score between kp_x and the most similar keyphrases can then be used as the system's confidence score.

Algorithm 1 Opinion History Creation

Input: D: Documents of known authorship
Output: H: Opinion history

procedure CreateOpinionHistory
    Foreach $d_i \in D$:
        $kp_i = extract\_keyphrase(d_i)$
        $we_i = extract\_word\_embeddings(kp_i)$
        $s_i = infer\_sentiment(d_i)$
        $H.append((kp_i, we_i, s_i))$
    End Foreach
    return H
end procedure

Sentiment extraction.

For every text that we've extracted the keyphrase of, we also extract the text sentiment. This way, the combination of the keyphrase/sentiment tells us what the author's opinion towards the topic of the text is. Sentiments extracted are in the form of a continuous value between -1 and 1, where the closer the value is to 1, the more positive the sentiment is, the closer it is to -1, the more negative the sentiment is, and the closer the value is to 0, the more neutral it is.

4.2.2 Application on the Text of Questioned Authorship

The final step of our model is to process a new text d_x of questioned authorship. After creating the opinion history of the suspected author A, we apply a similar procedure to d_x as the one we applied to every document when constructing the opinion history. We extract the keyphrase of d_x and calculate its word embedding we_x, and extract the sentiment s_x of d_x. Then, we run a similarity search algorithm, like cosine similarity between we_x and all the keyphrase embeddings in the opinion history to get the top n most similar keyphrases, which would reflect the top n most similar texts to d_x, in terms of context, which are published by A. Based on the sentiment of the top n most similar texts, we can predict what should the sentiment of d_x be. Let sp_x be that predicted sentiment. sp_x can then be compared with s_x, the real extracted sentiment of d_x, and based on a similarity threshold, the model can determine, with a certain confidence, if d_x was authored by A.

4.3 Implementation

For extracting keyphrases, we use KeyBERT [42], which is a keyword extraction technique that utilises BERT embeddings [104] to create keywords and keyphrases that are most similar to a document. For calculating the word embeddings of keyphrases, we use Albert [66] with the pre-trained English embeddings. For sentiment analysis, we use VADER Sentiment Analysis [52], a lexicon and rule-based sentiment analysis tool that specializes in inferring sentiments expressed in social media. And for clustering sentiments, we use DBSCAN algorithm [29]. All experiments were ran on Google Colaboratory[1]. Hardware Specifications:

- CPU: Intel(R) Xeon(R) CPU @ 2.30GHz
- GPU: Tesla P100-PCIE-16GB
- RAM: 26.3 GB

4.4 Experimentation

In this section, we discuss the experiments we did on our model, and how it compares to one of the top performing algorithms from the PAN ant CLEF 2015 author verification task, SPATIUM-L1. We also discuss the different hyper parameters in our model in need of tuning, and the methodology behind doing so. We also describe 2 datasets that we use: a long-text dataset which we use to confirm the performance of SPATIUM-L1 with the claims of the authors, and a short-text dataset composed of tweets which we use to firstly confirm that SPATIUM-L1 doesn't perform well on short text, and secondly to apply our approach on it.

4.4.1 Datasets

Long Text Dataset

For testing existing SPATIUM-L1, we use the PAN @ CLEF 2015 author verification task dataset, which is a training corpus composed of a set of problems, where each problem is described as a set of documents (1-10 per problem) belonging to a single known author, and exactly one document of questioned authorship. Within each problem. Each document

[1]https://colab.research.google.com

PAN 2015			
	Language	**Documents**	**Avg. words documents**
training	English	200	366
	Dutch	276	354
	Greek	393	678
	Spanish	500	954
testing	English	1000	536
	Dutch	452	360
	Greek	380	756
	Spanish	500	946

Table 4.1: Overview of the PAN 2015 authorship verification dataset (long text)

lengths vary from a few hundred to a few thousand words. An overview of the dataset is presented in Table 4.1

Short Text Dataset

To evaluate our approach, we use the dataset developed by Schwartz et al. [85] containing ∼9,000 Twitter users with up to 1,000 tweets each. We depart from the preprocessing followed by Schwartz, since we pereserve dates, times and references to other users (*@<user>*). We do this since in our case, we are interested in the author's opinion towards different keywords mentioned in their tweets. For example, if the main focus of a tweet is to talk negatively about someone, the author is likely to mention that someone's twitter user name, and that username is likely to be detected as the tweet's keyword. Or, the author might, for example, like a specific year model of a car, but dislike one from another year. However, we do remove the *@* sign from the beginning of the mention (*@<user>* becomes *<user>*), and we also replace web addresses with the meta tag *URL* as we don't see a contribution of such data to an author's opinion.

4.4.2 Evaluating SPATIUM-L1

SPATIUM-L1[2] is an unsupervised authorship verification model developed by Kocher and Savoy [60]. It was submitted to the PAN at CLEF 2015 Author Identification task, and placed 4th in the evaluation on English language. This approach is based on simple feature extraction and distance metric, where the top k most frequent terms are selected to verify the validity of the author. We run our experiments on this algorithm and the results are

[2]https://github.com/pan-webis-de/kocher16

displayed in table 4.2.

tweets per account	TP	TN	FP	FN	unknowns	c@1 score
10	50.0%	18.2%	5.5%	0%	31.8%	0.8388
20	34.9%	7.3%	4.7%	0%	52.7%	0.6458
50	9.1%	9.1%	4.5%	0%	78.2%	0.3223
100	6.8%	6.8%	3.9	0%	79.9%	0.2481
150	4.3%	5.4%	1.1%	1.1%	91.4%	0. 1821
PAN 15 Dataset	43.529%	10.588%	34.118%	2.353%	12.9%	0.6436

Table 4.2: SPATIUM-L1 performance on the short text dataset and the PAN 15 dataset. TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative

Replication on long text.

We have confirmed the results from the evaluation by testing the SPATIUM-L1 model on the PAN 15 AV dataset, and have gotten a **c@1** score of **0.6436** which is very close to the one reported at the PAN 15 evaluation of **0.689**. The results are shown in Table 4.2, row **PAN 15 Dataset**.

Experimentation on short text.

We run the SPATIUM-L1 algorithm on out short-text dataset. We vary the number of tweets per author used to infer the authorship of the tweet of unknown authorship, and we note that regardless of the number of tweets per account, the percentage of unknown outputs is high in comparison with that of the PAN 15 dataset. In addition, as the number of tweets per account increase, the **c@1** score decreases dramatically, and the number of unknowns increase greatly as well. This seams as a counter-intuitive behavior since one would expect better performance once more data is provided per account. The results of these experiments are shown in Table 4.2.

4.4.3 Evaluating The Semantic-Based Author Verifier

Parameters Tuning

After the selection of the top n most similar texts to d_x, we can predict what should the sentiment of d_x be by having an average weighted sum of the sentiments of the top n:

$$sp_x = \frac{\sum_{i=1}^{n} cos\_sim(we_i, we_x) \; . \; s_i}{n} \tag{4.1}$$

where sp_x represents the predicted sentiment of d_x, cos_sim is the cosine similarity function, and we_i and s_i represent the word embedding and sentiment of text i in the top n most similar texts respectively. This predicted sentiment can then be compared with s_x, the real extracted sentiment of d_x, and based on a similarity threshold, the model can determine, with a certain confidence, if d_x was authored by A. Algorithm 2 presents the authorship prediction process in this scenario.

Algorithm 2 Authorship Prediction

Input: d_x: Document of unknown authorship
H : Opinion history from Algorithm 1
Output: is_author: Is A the author, with confidence

procedure IsAuthor
    $kp_x = extract\_keyphrase(d_x)$
    $we_x = extract\_word\_embeddings(kp_x)$
    $s_x = infer\_sentiment(d_x)$
    $topn = get\_top\_n\_similar(kp_x, H)$
    $confidence = weighted\_average(topn)$
    $\tau = calc\_threshold(topn)$
    $is\_author = is\_within(s_x, \tau)$
  return $(is\_author,\ confidence)$
end procedure

Confidence and Threshold

Confidence. The confidence of our model is an estimate of its performance when predicting whether A is the true author of a document of questioned authorship d_x. Calculating the confidence is dependent on the similarity between the word embeddings of each of the top n similar keyphrases we_i in the opinion history and we_x, the word embedding of the keyphrase of d_x, which is the document of questioned authorship. The higher the similarity

between these embeddings (reflected by a cosine similarity value closer to 1), the higher is the confidence of the model's prediction. The confidence is calculated as the average of the similarities between we_x and each we_i (equation 4.2). This decision can be justified as follows: the closer we_x is to each we_i, the closer the topic of d_x is to the topics of previous statements by A. Hence, the more likely sentiment s_x is to be consistent with the sentiments s_i of the top n similar documents.

$$confidence = \frac{\sum_{i=1}^{n} similarity\_score_i}{n} \tag{4.2}$$

Threshold. The sentiment of an author might vary significantly with regards to the same topic; she might be consistent or inconsistent with her opinions. In a scenario where she is consistent, she might always have a relatively positive opinion about Apple phones for example, i.e: $0.7 \leq s_i \leq 1 \; for \; i \in [1, n]$. While in an inconsistent scenario, her opinion might vary significantly from one tweet to another, i.e: her top 5 similar tweets about Apple phones can have the sentiments: $[-0.7,\ 0.3,\ 0.8,\ 0,\ -0.4]$. This adds an additional layer of difficulty, because sp_x would be averaging a wide range of values, which would not be a good representative of the author's opinion. In the consistent scenario, her opinion average (sp_x) would be a good value to compare s_x to, since sp_x might be a value close to 0.8, so provided a certain *threshold* τ, we would just need to check if $s_x \in [sp_x - \tau,\ sp_x + \tau]$.

This problem can be solved by making the *threshold* adaptive to the spread of the sentiments s_i. We accomplish that by using the standard deviation of the sentiments (equation 4.3), where μ is the mean of the sentiments. The sentiment value is between -1 and 1. Thus, the value of σ is between 0 and 1. A σ value of 0 indicates that the sentiments are all equal; 0 spread.

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(s_i - \mu)^2} \tag{4.3}$$

And a σ value of 1 indicates maximum spread, that is: sentiments are equally divided between the exact values -1 and 1. Using this information, the threshold can be calculated as σ plus a leniency parameter α which can be tuned to suit different use cases: $\tau = \sigma + \alpha$. This provides a way to adapt our threshold to the spread of the sentiments.

We run our model on *dataset 1*. We use the tweets of 3000 authors, each having 1000 tweets. For each author, we use 910 tweets to build the opinion history, and create a validation set composed of 20 tweets: 10 belonging to the same author, and 10 randomly selected from other authors. For every author, we run the model and do predictions on the respective validation set. Each prediction produces a decision, if d_x was indeed published by the author or not, and a confidence score. We run 2 experiments to study the effect of the threshold τ

and n, the number of top similar word embeddings, on the performance of the model. In each experiment, we calculate the precision, recall and F1-score. For each experiment, we also vary the minimum confidence needed to consider a prediction as a valid prediction.

Varying the confidence. Varying the confidence is omitting any prediction with an associated confidence below a certain value. In our case, increasing the minimum required confidence improves the precision on the expense of recall. The results are shown in Figures 4.4 and 4.5. When no minimum confidence was required, the model had a precision of 0.53 and an F1-score of 0.26. Increasing the minimum required confidence drastically changes the model's performance when increasing n and τ, reaching a precision of 0.92 while the f1-score suffers due to low recall, the f1-score and recall have a value of 0.02.

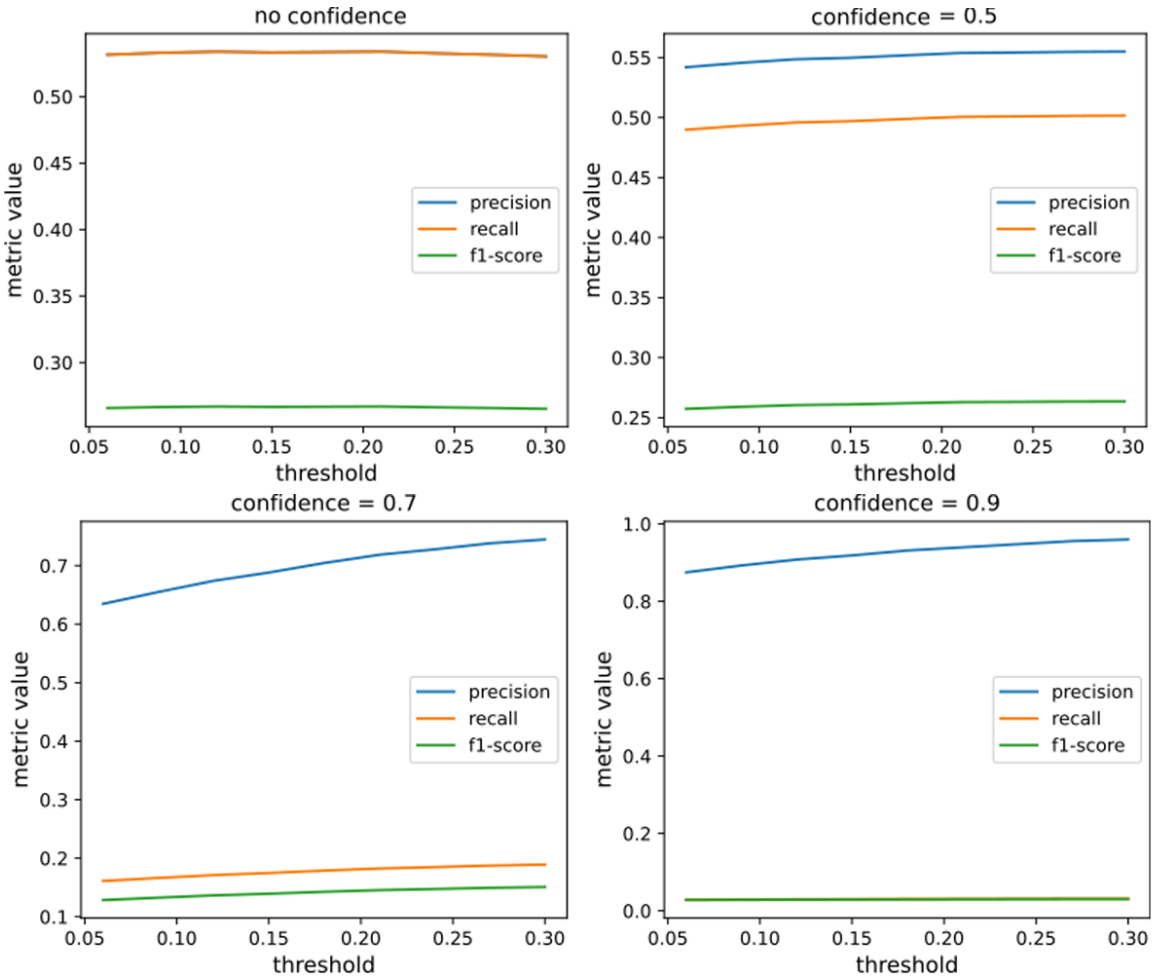

Figure 4.4: Model performance as a function of $threshold$ with varying confidence

Varying τ. τ affects the comparison margin between s_x and sp_x. τ is altered by changing the leniency parameter α. We variate α with the following values: 0.06, 0.09, 0.12, 0.15, 0.18,

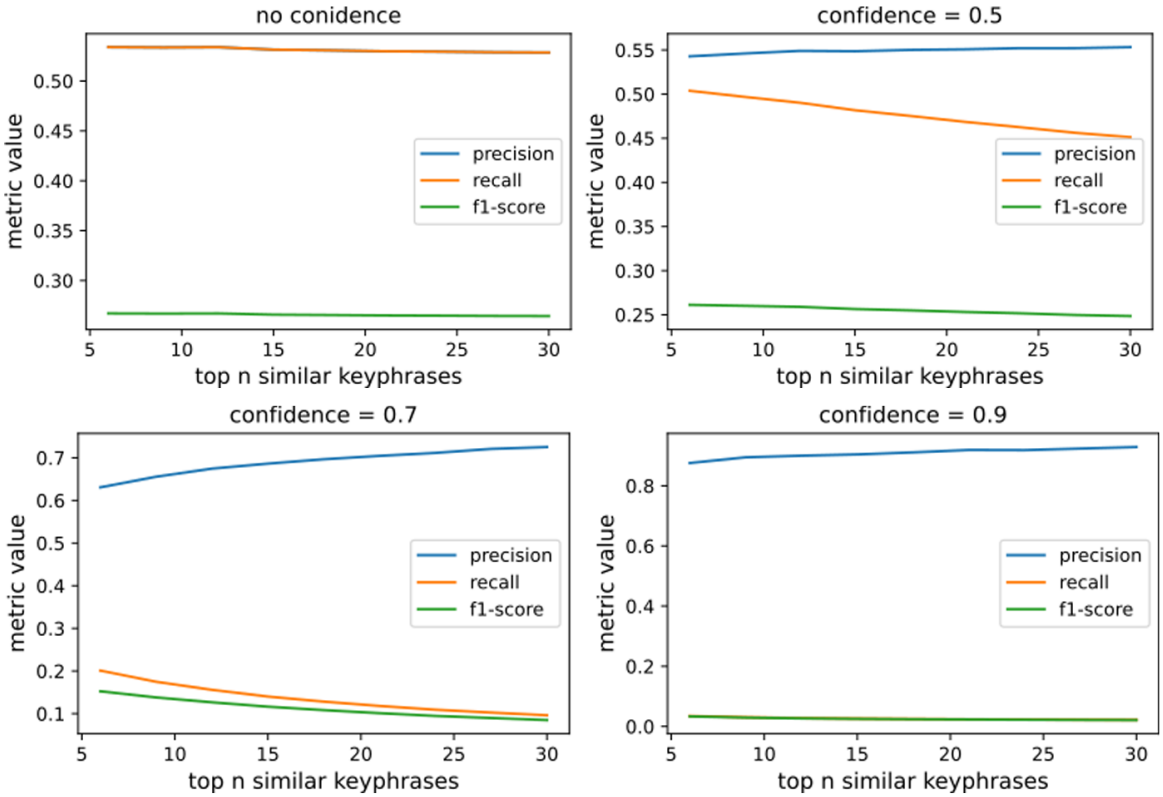

Figure 4.5: Model performance as a function of n with varying confidence

0.21, 0.24, 0.27, 0.3. The results of this experiment are shown in Figure 4.4. We notice that for all confidence values, precision, recall and consequently, f1-score increase as we increase τ. This is because it adds more tolerance for accepting s_x values. However, increasing τ too much would result in a decrease in precision, as more and more s_x values will be accepted.

Varying n. Changing the value of n adds more documents of relevance from the opinion history to compare with d_x. We varied n with the following values: 6, 9, 12, 15, 18, 21, 24, 27, 30. The results of this experiment are shown in Figure 4.5. We notice that when applying confidence restrictions, the precision increases with the increase of n, on the expense of recall. Precision increases as we increase n and the confidence restriction until it reaches 0.92, for $n = 30$ and confidence = 0.9, while recall suffers drastically, decreasing from 0.52 to 0.02, and consequently, the f1-score drops to 0.02 as well. This is because having additional documents of relevance narrows down the range of accepted sentiments.

Discussion

Our model produces results of high accuracy when choosing high minimum confidence, and high top n. This implies that, given enough documents of high relevance to a topic, we indeed do capture the opinion of an author A towards that topic. The implication being, acquiring enough documents centered around said topic to make an accurate decision. However, we do similarity search on word embeddings that take the semantic aspect of keyphrases into account. So the topics of the documents need not to be of an exact match to that of s_x. In addition, the same logic can be applied when facing a new topic never encountered before; the model can find the closest topic, semantically, within a certain level of confidence, and estimate the opinion accordingly.

Chapter 5

Improving Semantic-Based Author Verification and Impersonation Attacks

In this Chapter, we will discuss changes and improvements done to our semantic-based author verifier. We apply a clustering technique to the sentiments for the verification of true authorship, and introduce the different manners in which we tune the hyper parameters. We also introduce a new dataset of twitter influencers, which we've collected for the purposes of testing our model. In addition, we test Adominem [11], a siamese network for representation learning, and the top performer at the PAN 2020 AV task [54], which focuses on AV in short text, and compare its performance on our Twitter influencers dataset. Finally, we apply doctored manipulation to 100 tweets in a manner that changes the semantics, and show that our approach can detect manipulated documents with high accuracy, and compare that to the performance of Adominem in the same task.

5.1 Influencers Dataset

For the purposes of this work, we develop a new Twitter dataset formed from the most recent tweets of the top 88 most followed Twitter users. We start from the idea that an impersonator - someone who wants to use others' accounts to spread information, usually with bad intentions- would be more likely to impersonate an online influencer than a user with a low online presence. Thus, we've collected tweets of the top 100 most followed Twitter users. Due to the twitter API limitations, we were only able to fetch the 3200 most recent tweets per author, which should still provide enough data for analysis. After removing non-English accounts, neutral accounts (for example, sports updates accounts, like @NBA) and twitter accounts with less than 500 tweets, we ended up with 88 twitter accounts. This might not seem like a lot of accounts, compared to, for example, the 9000 twitter accounts collected by [85], but we argue that this is enough to evaluate our model, since it works on a per-author basis.

For the data preprocessing, we follow the same procedure as we did for the short text dataset in Chapter 4. We pereserve dates, times and references to other users (*@<user>*), remove the *@* sign from the beginning of the mention, and we also replace web addresses with the meta tag *URL*. The rest of the text is maintained as is. An example of a pre-processed tweet is presented in Table 5.1.

Original Text	**Pre-processed Text**
@Erdayastronaut @flcnhvy @SpaceX Yeah, will take less than a minute to order on https://t.co/Q1VvqVmJ2i when it goes live	Erdayastronaut flcnhvy SpaceX Yeah will take less than a minute to order on URL when it goes live

Table 5.1: Example of tweets pre-processing

5.2 Semantic-Based Author Verifier With Anomaly Detection

In this technique, we model the AV problem as an anomaly detection problem, where we check the sentiments of the documents of questioned authorship to see if they fall within the norms of A for a given topic, or they register as an anomaly. We utilize DBSCAN [29], a clustering algorithm, as an anomaly detection tool. It clusters data points into groups, and the points which do not fit in a group (anomalies) are labeled as -1. Algorithm 3 presents the authorship prediction process through anomaly detection of sentiments.

Algorithm 3 Authorship Prediction Through Anomaly Detection

Input: d_x: Document of unknown authorship
H : Opinion history from Algorithm 1
ϵ : epsilon parameter of DBSCAN
$min\_samples$: minimum samples parameter of DBSCAN
Output: is_author: Is A the author

procedure AnomalousIsAuthor
    $kp_x = extract\_keyphrase(d_x)$
    $we_x = extract\_word\_embeddings(kp_x)$
    $s_x = infer\_sentiment(d_x)$
    $topn = get\_top\_n\_similar(kp_x, H)$
    $sentiments = get\_sentiments(topn)$
    $clusters = DBSCAN([sentiments, s_x], \epsilon, min\_samples)$
    $is\_author = -1 \neq clusters[s_x]$
  return is_author
end procedure

There are two parameters of DBSCAN which we need to shed light on:

1. Epsilon ϵ: data points would need to be within distance ϵ to belong to the same group.
2. Minimum samples $min\_samples$: the minimum number of data points within distance ϵ required to form a dense region.

The values chosen for these parameters have a huge effect on the performance of our model. In addition to these parameters, we have: n and min_sim. min_sim is an optional parameter which we can use to force a **min**imum **sim**ilarity measure between the top n matching documents and d_x for them to be selected for analysis. So min_sim ranges between 0 and 1, where a higher value forces to select documents with higher similarity measures to d_x, which would increase the model's confidence and potentially increase the accuracy, on the expense of recall, since more cases would not be answered.

We distinguish two different approaches for tuning parameters: $Static$ and $Dynamic$. Regardless of the approach, ϵ can be tuned separately. Figure 5.1 (a) shows the performance variation of the model as a function of ϵ. Too high of a value, and the model would not be able to detect outliers. Too small, and it would have a lot of False Negatives (true authors identified as impostors). As a balance between precision and recall, for the rest of the experimentation, we set the value of ϵ to 0.15.

5.2.1 Parameter Tuning

Static Tuning

In static tuning, we fix the value of n and $\min\_samples$. A higher n means that more documents of matching topic will be selected for analysis. This would increase the accuracy in the case where the selected documents have a high similarity values. But if n is too high, the more documents with low topic similarity will be selected, which would dilute the final result. Too low, and we would over-fit the model's decision based on a few documents. Figure 5.1 (b) shows the variation of the model's performance as a function of n, with $\epsilon = 0.15$ and $min\_samples = 5$.

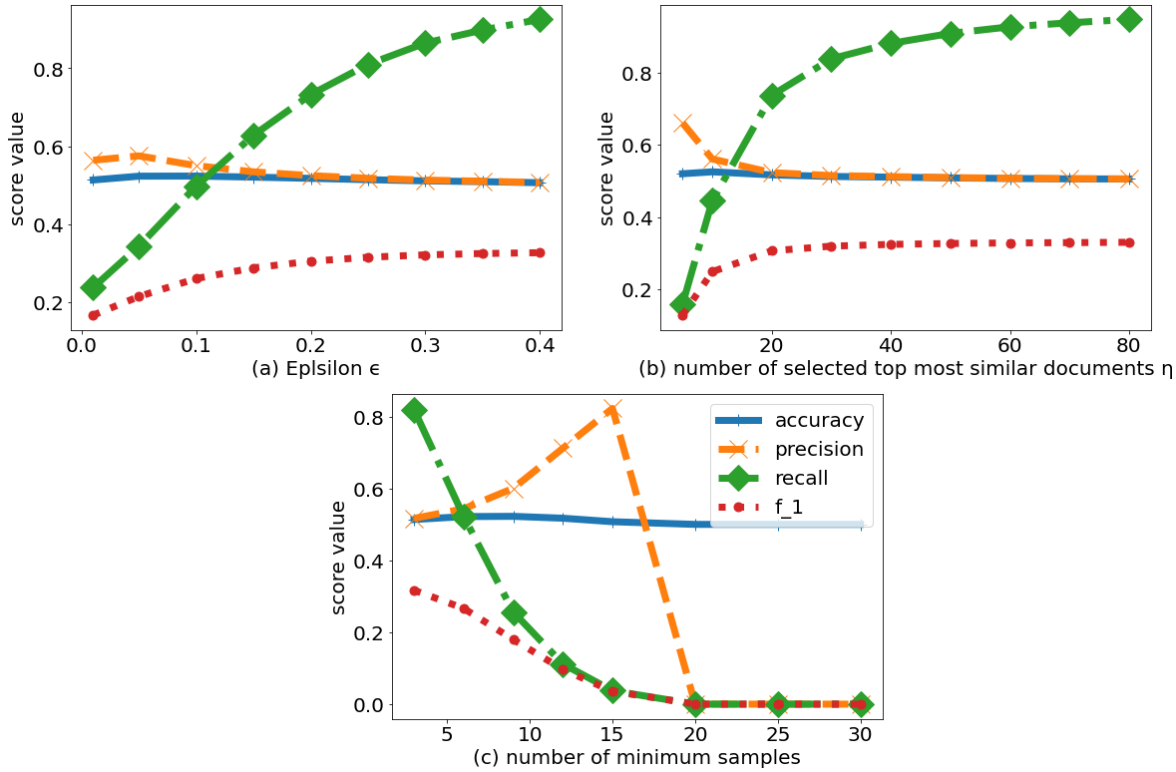

Figure 5.1: Initial model performance with static tuning.

A higher $min\_samples$ means that more data points within close distance (governed by ϵ) will be needed for a data point not to be an outlier. This translates to the model needing a larger number of documents matching the sentiment of d_x for it not to be an outlier. Note that $min\_samples$ needs to be $\geq n$, since then there would not be enough data points to cluster a dense group. Figure 5.1 (c) shows the variation of the model's performance as a function of $min\_samples$, with $\epsilon = 0.15$ and $n = 15$. Note how all the performance drops

to zero when $min\_samples$ is greater than $n = 15$.

Choosing a value for min_sim plays a huge role on the model's performance. The higher min_sim is, the more relevant the documents chosen from OH are to d_x. However, if the documents of highest similarity score are less than min_sim, the model returns no answer. We note a performance comparison that needs to be made of the model's performance as a function of n for different min_sim values, since the 2 parameters are related. This is illustrated in Figure 5.2 where we note that as min_sim increases, the number of unanswered queries increase. However, we also note that the number of incorrect answers drops to almost zero with the increase of n when $\min\_sim$ is high.

5.2.2 Dynamic tuning

Motivated by the results from the static tuning, we decided to make further experiments that take advantage of min_sim. Thus, with dynamic tuning, we only set the value of min_sim, and that in turn sets the value of n as the number of matching documents with similarity score $\geq min\_sim$. However, as $min\_samples$ is directly affected by min_sim, we also dynamically set the value of $min\_samples$:

$$min\_samples = \left\{ \begin{array}{ll} 3 & \text{if } n \leq 10 \\ floor(\frac{n}{3}) & \text{if } n > 10 \end{array} \right\}$$

We set $min\_samples$ to 3 if n is less than 10 because of the results we have in Figure 5.1 c. And for $n > 10$, we basically allow for a maximum of 3 possible groups of clustering, for the cases where the author might have a diverse group of opinions.

As per the model's confidence score, there are 2 factors at play here: n and the similarity score of each selected document. We need an algorithm that provides a balance between how big/small n is, and the similarity scores of the selected documents. So instead of averaging the similarity scores, we pass their sum through an altered version of the sigmoid function represented in Equation 5.1.

$$confidence = \frac{2}{1 + e^{(l.c)}} - 1 \tag{5.1}$$

Where c is the sum of all similarity scores of the selected n documents, and l is a parameter that governs how quickly or slowly the confidence score converges to 1 as a function of c. Figure 5.3 shows the effect of min_sim on the performance of the model as a function of l. We choose to represent the performance in terms of true positives, true negatives, false

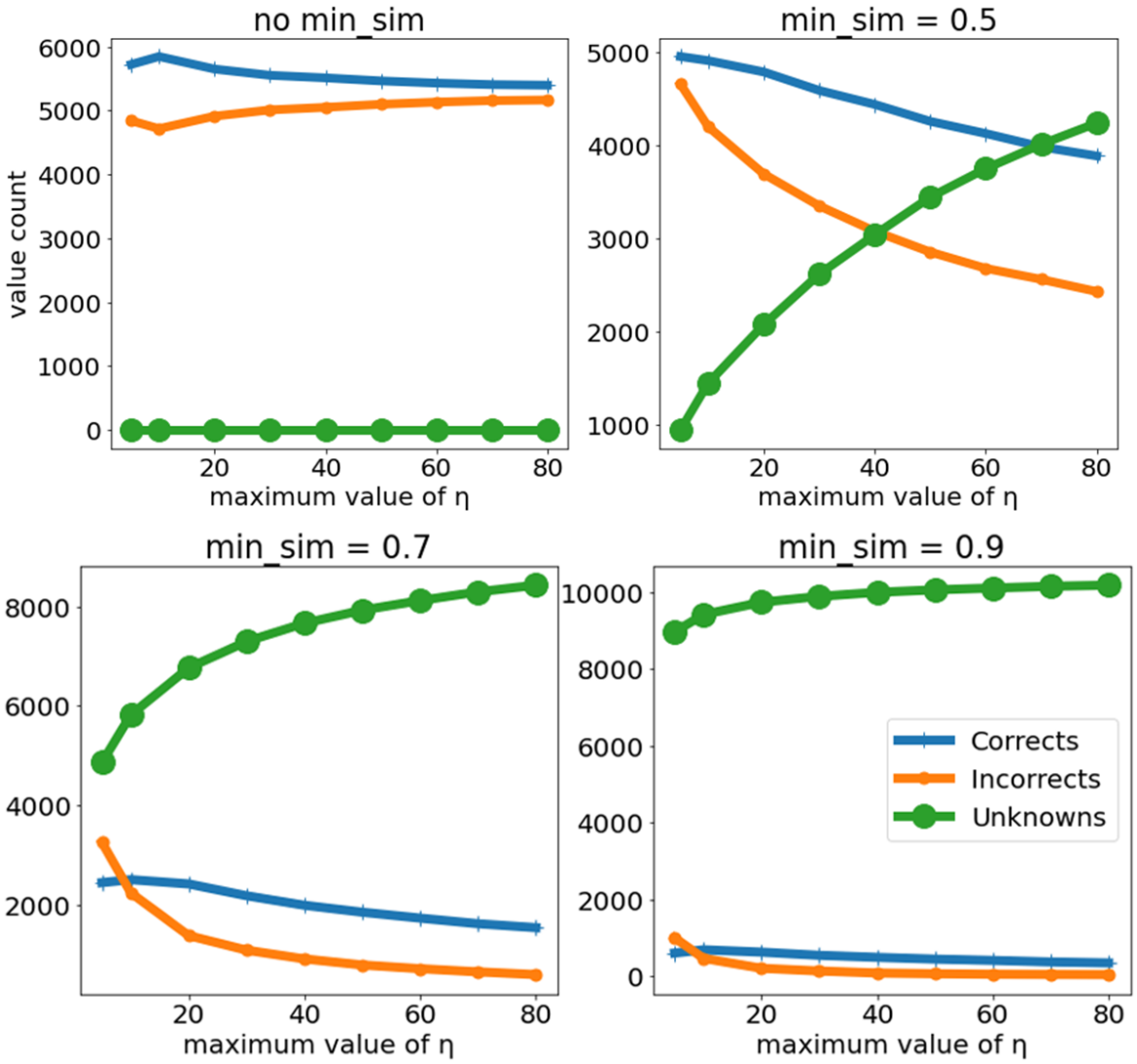

Figure 5.2: Model performance with static tuning as a function of n for different values of min_sim.

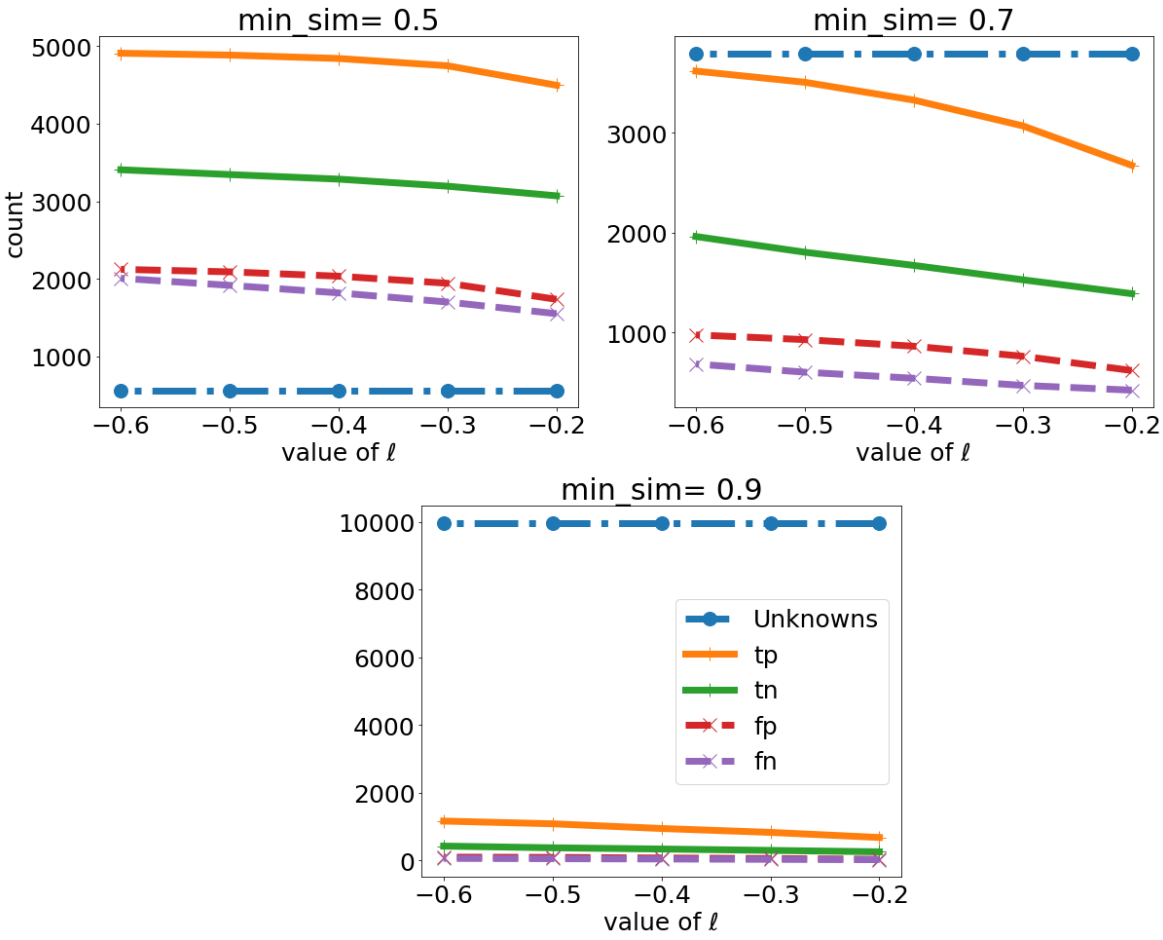

Figure 5.3: Model performance with dynamic tuning.

positives and false negatives. We can see how the number of unknown results increase for higher min_sim values (0.7 and 0.9), but that is accommodated by a drop in the number of false negatives and false positives.

5.3 Evaluating Adhominem

Adominem [11] is a siamese network for representation learning, and is the top performer at the PAN 2020 AV task [54], which focuses on AV in short text. It makes a great candidate to test against micro-messages, since it was specifically designed to tackle AV in short text. To train Adhominem, we provide a file where each line contains 2 tweets, labeled by 1 (the 2 tweets are authored by the same author) or 0 (the 2 tweets are **not** authored by the same author). We trained Adhominem on a subset of our tweets dataset composed of 40,000 lines (80,000 tweets) for 20 epochs. Then we evaluated performance on three subsets of tweets:

1. Testing subset: is the testing set of the training dataset (20%), composed of 8,000 lines.
2. Validation subset: a set of 1,000 lines unseen during training.

The results of the experiment on these three subsets are shown in Table 5.2.

Dataset	Acc	c@1	F1
Testing	70.4	0.719	0.758
Validation	56.2	0.522	0.624

Table 5.2: Adhominem performance measurement on different micro-messages datasets.

Discussion

Testing and validation datasets. Adhominem reportedly had an overall performance score of 0.935 in the PAN 2020 AV task, which was aimed at AV in short texts. However, the short texts handled there were anecdote snippets on the order of over a hundred words per snippet, while micro-messages are on the order of tens of words. This creates a great deficiency in features to learn, which decreases the model's performance scores. In addition, Adhominem's performance suffers even more when exposed to a yet unseen validation dataset. We argue that this is due to the model over-fitting on the training set.

5.4 Impersonation Attacks

In this Section, we discuss a final set of tests that we ran which are on a manipulated dataset, where we doctored a subset of tweets in a manner that changes the semantics without changing the writing style. The aim here is to show a weakness point present in current AV approaches, which is that they don't take semantics into account, which leaves them vulnerable to manipulated documents that preserve the stylistic features of the author. We run these tests on both our model, and on the state of the art Adhominem for comparison.

5.4.1 The Manipulated Dataset

We've selected and altered a set of 100 tweets from the influencers dataset in a way that preserves the writing style while changing the meaning behind it. All the tweets are labeled 0, meaning they are not authored by the real author, since they are all manipulated messages. A sample of this dataset is shown in Table 5.3.

Original tweet	Altered Tweet
What a **glorious** place New Zealand is. **Great** to be back	What a **terrible** place New Zealand is. **Sad** to be back.
gagosian art opening for Harmony**!!** One of the **most influential**. So proud	gagosian art opening for Harmony, one of the **ugliest**, so **upset**

Table 5.3: Sample tweets from the manipulated dataset.

5.4.2 Adhominem on The Manipulated Dataset

Adhominem had an accuracy score of 36.2. Note that on the manipulated dataset, the model's job here is to infer that all the data in the manipulated dataset is of false authorship. Since in this case there is no true positives, only true negatives and false negatives, F1 score will always be 0. So here we can only calculate accuracy, which is the number of texts correctly identified as false authors.

5.4.3 Our Model on The Manipulated Dataset

Our results here show a huge potential for manipulated text detection. our model has an accuracy of 93% on the manipulated dataset, which greatly outperforms Adhominem on the manipulated dataset with a accuracy of 36.2%. This shows that taking the semantics into consideration when performing AV can have a substantial effect in detecting online impostors pretending to write in the same manner as an author A.

As it is the case with Adhominem's performance measure on the manipulated dataset, we cannot measure the F1 score for our model here.

5.4.4 Discussion

We started this work with two hypotheses. The first is that stylistic-based approaches trained on short text don't perform well when applied on micro-messages, which we have proven in the results of Adhominem on the validation dataset. The second hypothesis is that stylistic-based approaches don't take text semantics into consideration. We argue that we've shown this to be correct with the results of Adhominem on the manipulated dataset. Although a dataset of 100 entries is not sufficient to make a deduction, it does provide an intuition on the overall potential performance of the model, especially considering the vast difference in performance between the results of the validation dataset and the manipulated dataset.

Chapter 6

Temporal Effect of Data in Author Verification

6.1 The Temporal Influence on Author Verification

In Chapters 4 and 5, we studied the problem of author verification in micro-messages by analyzing an author's opinions towards certain topics, and later used that to determine whether or not they are indeed the author of a document of questioned authorship, by matching the topic and opinion of said document to that of the author. A problem with this approach is that people tend to change their opinions over time. An author who is well known to like Apple phones might change their mind if Apple released a bad phone, or if said author was introduced to a phone they like that was not manufactured by Apple.

Thus, it is important to study authors' opinion change over time. That is to say, study the temporal effect of data on authorship verification. This in itself could be quite a challenging problem due to the following reasons:

- Each author would have their own pace of opinion variation over time
- Within one author's opinion history, different topics would have different opinion variations over time

So opinion variation could be studied on an individual level, where a single author's publications, and their associated topics, are collected over time, and their associated opinions are studied. It could also be studied as a global scenario, where external effects (political, cultural, ...) would have an impact on an entire population group. In this case, we can see trends in opinion shifts. Gabel et al. [35] work on such a scenario, where they study the influence of elites' messages on the political opinions of mass populations. They wanted to answer the question: *"Do elite communications lead public opinions?"*. They propose an identification strategy to estimate the causal effect of elite messages on public support for the employment of changes in political institutions by the European integration. Their results show that more negative elite messages about European integration do indeed decrease public support for Europe. However, their study finds a consistency in opinions for more politically aware individuals. So elite messages have varying degrees of influence on the individual level, based on the background knowledge of each individual. But, on a more global scale, there would be a noticeable shift in opinion.

On the direct topic of author verification, Van Dam et al. [19] investigate the influence of topic and time on AV accuracy. Regarding topic influence, they found that cases with documents of similar topics overall (positive and negative) were found to increase accuracy of AV, while using short and diverse sets of reference documents, spanning multiple topics, the authorship verification problem becomes more difficult and the author verification accuracy drops in comparison to single-topic long reference documents.

Test set	**# Test cases**	**Accuracy**
All annotated	1368	0.633
Similar (<1wk)	684	0.665
Different (>3yr)	684	0.588

Table 6.1: The recorded accuracy when comparing comments made within one week to comments made more than three years apart [19]

As for the influence of time, Van Dam et al. found that authors do change their *writing style* over time. They compared Wikipedia Talkpages contributions made within a week with Wikipedia Talkpages contributions made years apart. The results of their findings regarding the temporal effect are displayed in Table 6.1. The accuracy of the Similar test set, written less than a week apart (0.67) is considerably higher than the accuracy of the Different test set, written more than 3 years apart (0.58).

Motivated by the above studies, in this Chapter we further investigate the extent of authors' opinion variation per topic over time, and the consequences of such variation on opinion-based author verification. As a starting point, we develop an artificial dataset of opinions to visualize opinion variations and get a better understanding of it. We then model the temporal effect problem as a classification problem, extended from our work in Chapter 5. Finally, we apply it to our dataset of influencers, described also in Chapter 5.

6.2 Problem Simulation

6.2.1 Artificial Opinion Data Generation

In order to get a better understanding of the problem, we start with an ideal scenario. We develop an artificial dataset of opinions where we assume that the entire dataset belongs to a single author, and references a single topic. To generate this artificial dataset, we run some estimations to make it more realistic in terms of volume of tweets per user.

There were 186 million twitter users in 2020 according to the Business of Apps[1], with 500 million tweets sent per day according to David Sauce[2]. This averages to 2.68 tweets per day per user in 2020. So as an average, our artificial dataset must have a time distance of 32238 seconds (or 8 hours and 57 minutes) between tweets. Since we are only interested in the tweet opinion and date of publish, our generated data contains only these 2 attributes, as shown in the sample below:

```
temporal_opinion_data_sample = [
  {
    "timestamp" : 1625220635,
    "sentiment" : 0.6,
  }
]
```

[1]https://www.businessofapps.com/data/twitter-statistics
[2]https://www.dsayce.com/social-media/tweets-day

We also take into account making gradual shifts in opinions over time. This is to reflect that sentiments of the tweets will not make a sudden drastic jump from, say, being negative to being positive. So if the author had a positive opinion about a topic, they would slowly shift into negativity over time. The code that we've created to generate the sample data is shown in Algorithm 4, and the data is visualized in Figure 6.1. We generated 5000 data points, reflecting 5.1 years of an average person's twitter life. In addition, the Figure also contains a data point related to a document of questioned authorship, on which we will apply our analysis.

From Figure 6.1, notice that the opinion of the author A is shifting towards being negative starting the end of 2016, while the document of questioned authorship d_x has a positive sentiment (0.65). In the time window of d_x (early 2017), A has an almost neutral opinion. Without considering the temporal aspect, our approach in Chapter 5 would classify d_x as being authored by A, since it averages out all the opinions, regardless of what time they were published at.

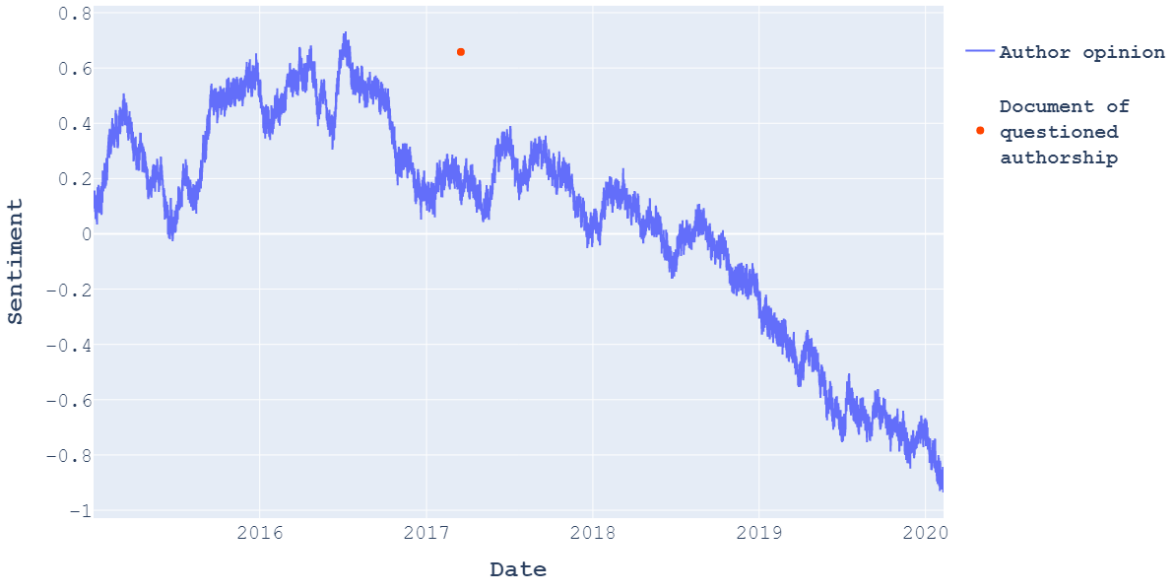

Figure 6.1: Artificially generated opinion data over time

6.2.2 Application on the Artificial Dataset

Following the work done in Chapter 5, we can still model this problem as a clustering problem. However, given d_x, we no longer consider the entire dataset for analysis, or at least, all the documents in D having a similar topic as d_x. Rather, we take the subset of D

Algorithm 4 Artificial opinion data generation

Input: $startDate$: The date from which the data points begin
n: The number of data points to make
Output: $opinionSamples$: A list of the generated opinions

```
1:  procedure GenerateArtificialOpinions(startDate, n)
2:      timeDistance ← 32238                          ▷ The time distance of 32238 seconds
3:      lower ← 0.1                                   ▷ The ranges by which we can change the sentiment
4:      upper ← 0.2
5:      lowerBound ← −1                               ▷ The lower and upper bounds of sentiment
6:      upperBounds ← 1
7:      for period ← 0 to n do
8:          startDate ← startDate + timeDistance
9:          if Random is 0 then                       ▷ Randomly select to increase sentiment
10:             if upper + 0.01 < upperBound then     ▷ Increase bounds
11:                 lower ← lower + 0.01
12:                 upper ← upper + 0.01
13:                                                   ▷ Else bounds stay the same
14:             end if
15:         else                                      ▷ Randomly select to decrease sentiment
16:             if lower - 0.01 > lowerBound then     ▷ Decrease bounds
17:                 lower -= 0.01
18:                 upper -= 0.01
19:                                                   ▷ Else bounds stay the same
20:             end if
21:         end if
22:         opinionSamples.append({
23:             "timestamp" : startDate,
24:             "sentiment" : GenerateRandomeNumber(lower, upper, 1)
25:          })
26:     end for
27:     return opinionSamples
28: end procedure
```

where the documents have a similar topic as d_x, and also fall within a certain time window from d_x. To accomplish this, we can isolate the author's opinions related to the topic of d_x within a certain time window of d_x, as depicted in Figure 6.2, and run the clustering algorithm on them.

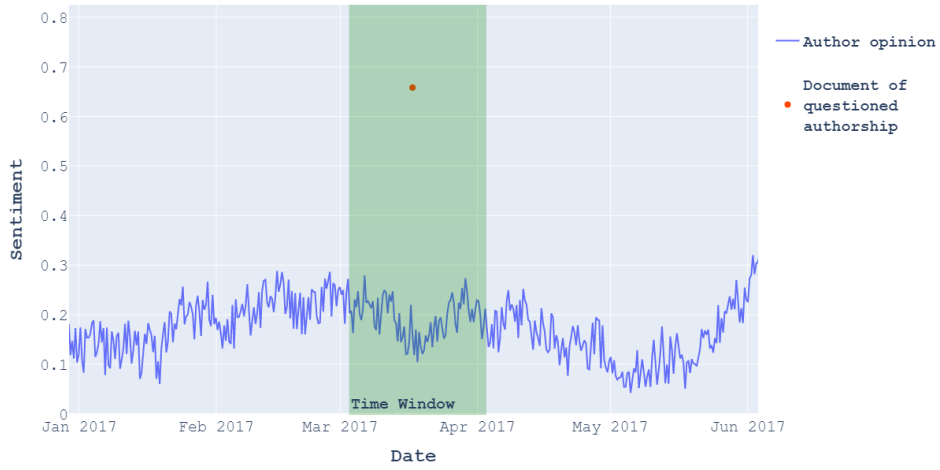

Figure 6.2: Isolated document of questioned authorship within a time window

In this scenario, we select a time window of 1 month, from *18:30:36 02-03-2017* to *05:35:42 04-04-2017*, and we get 88 author opinions to use. We run DBSCAN on these opinions, with the opinion of d_x appended, with the parameters: $\epsilon = 0.05$ and $min\_samples = 10$. The results are shown in Figure 6.3, where the classification algorithm was able to successfully identify the sentiment of d_x as an anomaly. Although this seems as a trivial example, but we can apply the same procedure on our influencers dataset to check for real-life results.

6.2.3 Time Window Size

In order to get further insight on the effects of the time window size on the accuracy of the clustering process, we increase the number of documents of unknown authorship. We add 9% of the initial size of our artificial dataset of sentiments as outliers. This new noisy dataset is presented in Figure 6.4. A perfect clustering would detect that all additional sentiments of documents of unknown authorship are outliers.

We run the clustering on the noisy artificial dataset, and we vary the window size on each run, from 2 days to 1000 days. An n-day time window size means n/2 day to the left of the day of publishing of the document of unknown authorship, and n/2 days to the right. The results are shown in Figure 6.5. Since all added sentiments should be classified as outliers,

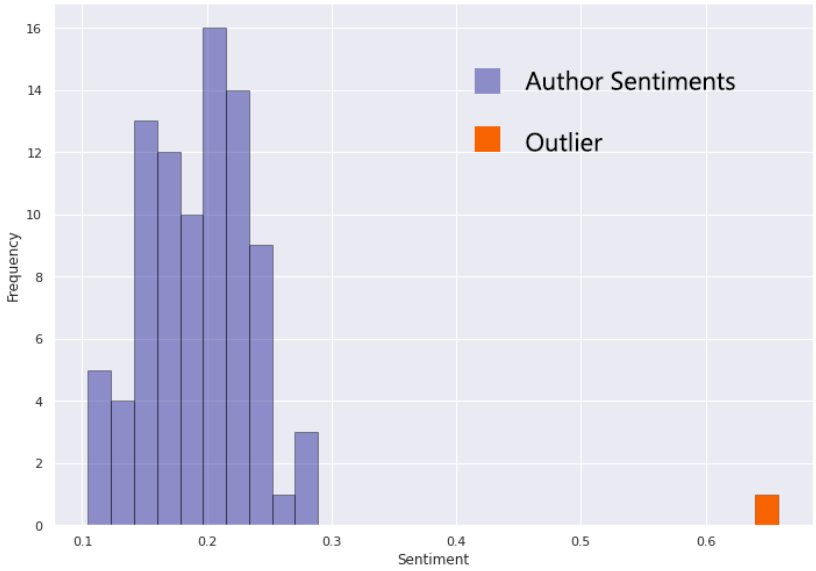

Figure 6.3: Clustering results of documents sentiments within d_x's time window

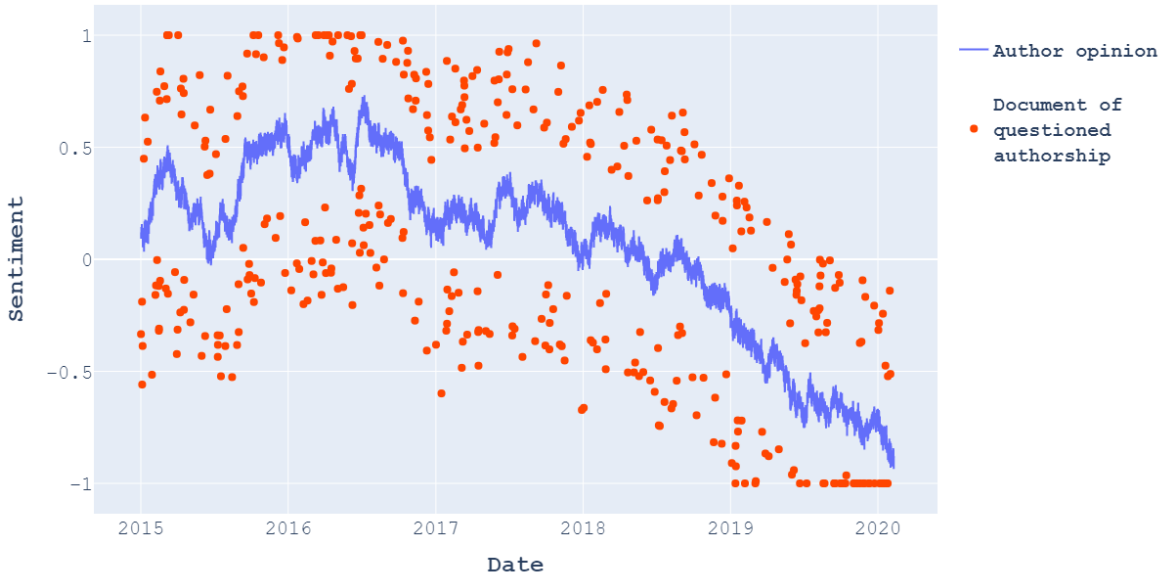

Figure 6.4: Artificially generated opinion data over time with added sentiments of different authorship

we only analyze the performance in terms of true positives and true negatives. We can see from the results that all outliers were detected until a window size of 102 days, which is when true negatives started increasing, and true positives started decreasing with the increase of the size of the time window. This is because the bigger the time window is, the more opinions will be taken into consideration when detecting outliers. This introduces past/future opinions which may not conform with the author's opinion on the time of the document of questioned authorship, causing what should be an outlier to be conceived as a genuine data point.

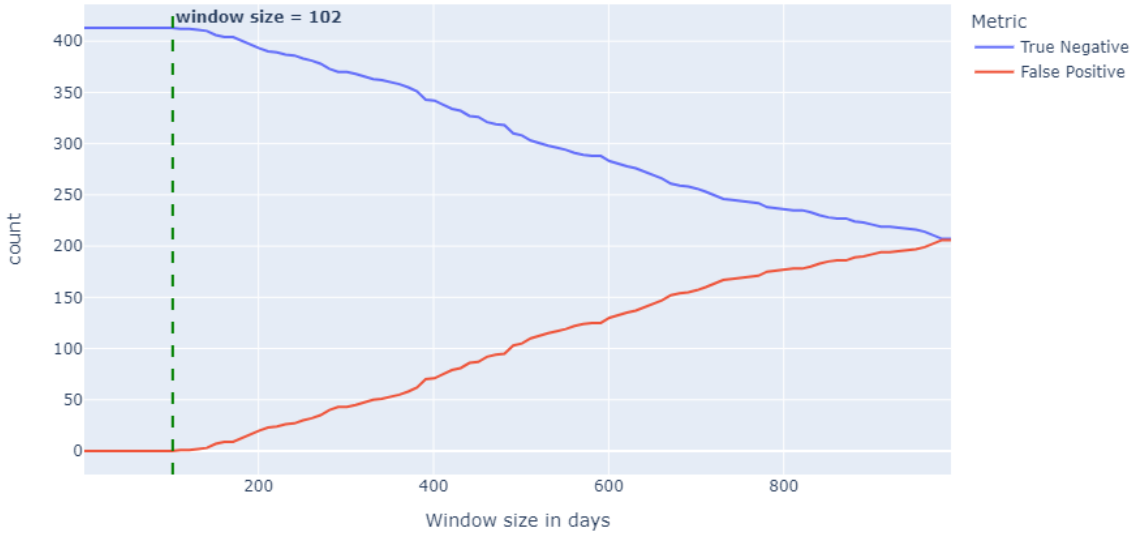

Figure 6.5: Clustering performance on the noisy artificial opinion data as a function of time window size

This opened the way to analyzing how big the time window should be with respect to the data at hand. In order to determine the size of the time window, we need to consider 2 aspects: how dense are the sentiments (how many sentiments in the specified time window) and how spread out the sentiments are (how much they vary in positiveness and negativeness. i.e, the standard deviation). These 2 aspects reflect nicely with the parameters $min\_samples$ and ϵ from the DBSCAN algorithm. The tuning of the DBSCAN parameters was discussed and experimented with in Chapter 5, but here we discuss it while taking the temporal aspect in perspective.

6.3 Application on The Influencers Dataset

In this section, we will discuss the steps taken to apply the temporal procedure discussed above on the influencers dataset that we have introduced in Chapter 5, and analyze the results.

6.3.1 Data Selection

Collecting data from each author that belong to a unified topic, and span a relatively short period of time is a challenging task. Our dataset is only collected from tweets, while in a real-life scenario, investigators could collect from any source of information that an author was known to publish in, either in textual or verbal contexts. Thus, we were not able to run an experiment on a large number of topics. We ran our experiment on the topic of **Apple**, the tech company, collecting tweets from all authors that relate to Apple, and then inserting a new document that could have either originated from the author themself, or was of a fabricated origin. The tweet's topic must have a minimum cosine similarity of 0.6 for the tweet to be chosen.

After applying the constraints to the influencers dataset, 11 accounts met the conditions. Figure 6.6 displays the selected accounts, the distribution of their sentiments, as well as that of the inserted document belonging to the author, and of false authorship.

6.3.2 Parameter Tuning

min_samples. $min\_samples$ is number of opinions in a neighborhood for a point to be considered as a core point. We can use this to dynamically select the time window size by expanding the time window until a minimum number of opinions (ρ), proportional to $min\_samples$, has been selected. We achieve this by iterating over a range of values for $min\_samples$, from 3 to 12, and for each value of $min\_samples$, we also iterate over a range of ρ equal to $n.min\_samples$, with $1 \leq n \leq 10$ to check how the selection of minimum number of opinions affect the performance.

Epsilon ϵ. ϵ reflects the distance between two opinions for one to be considered as in the neighborhood of the other. Thus, in the case where the author's opinions are more spread out, a bigger epsilon value is needed to accommodate. To achieve this, we give ϵ the value of the standard deviation of the sentiment of the opinions within the selected time window, after passing it through an modified sigmoid function $mod\_sigmoid$, presented in equation

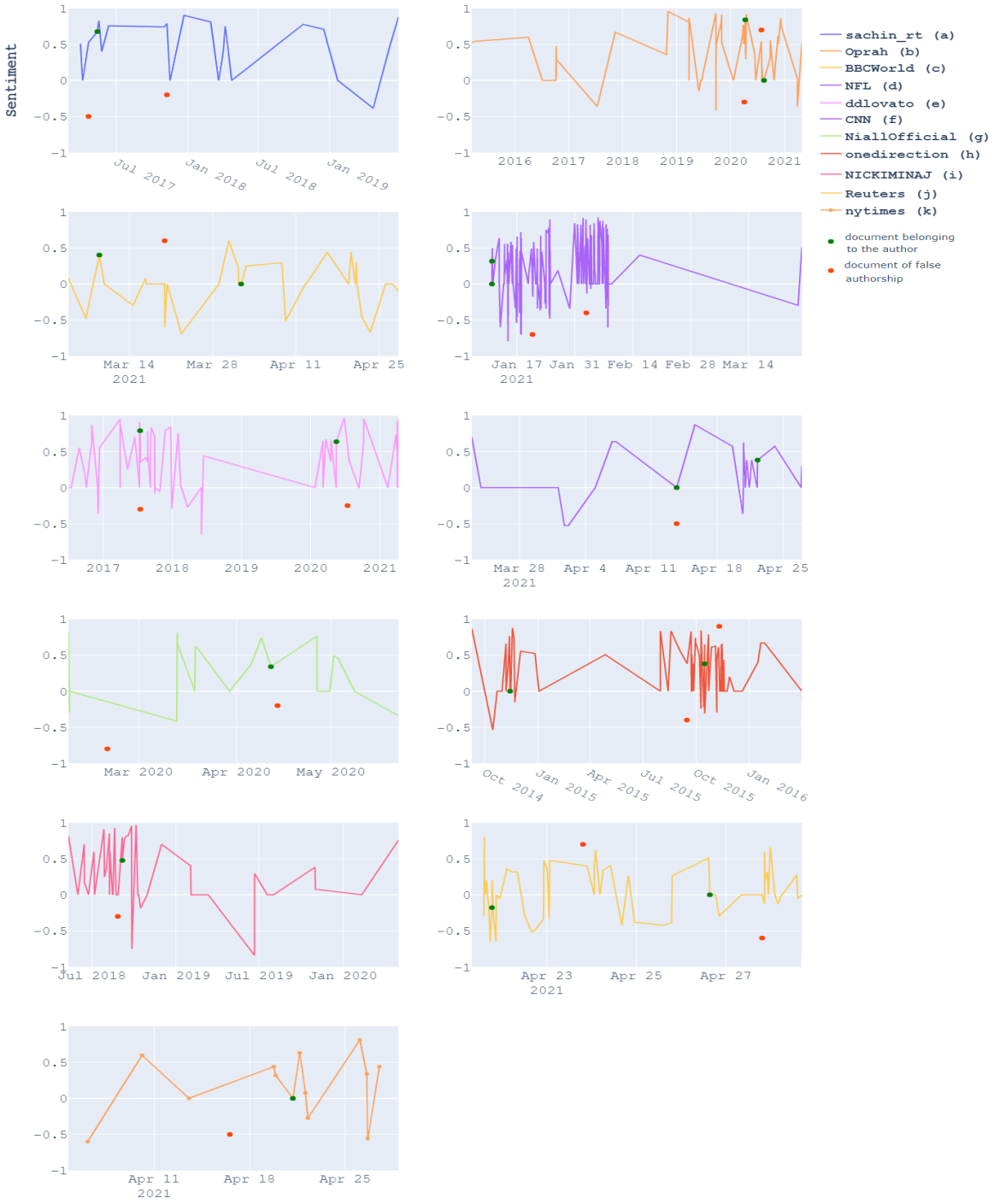

Figure 6.6: Sentiment distribution of the selected accounts from the influencers dataset, and the inserted document of true and questioned authorship

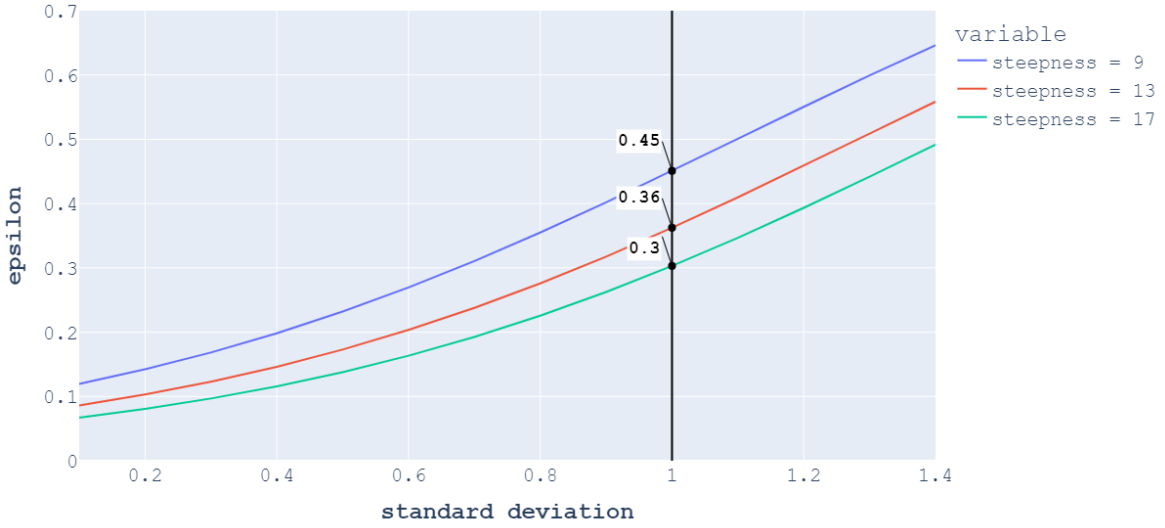

Figure 6.7: The function $mod\_sigmoid$ applied to the standard deviation with different values of $steepness$

6.1.

$$mod\_sigmoid = \frac{1}{1 + steepness.e^{-2x}} \tag{6.1}$$

$mod\_sigmoid$ gives a gradual increase of ϵ with the increase of the standard deviation, since considering our experiments with Chapter 5, a relatively small value of ϵ would produce the better results. So if sentiments are close, (standard deviation = 0.2, for example), we would want ϵ to be around 0.05, and a similar case with more spread out sentiments, where we would want epsilon to be around 0.4. In addition, $mod\_sigmoid$ also has a parameter $steepness$ that dictates how quickly the output increases with the increase of the standard deviation. The output of $mod\_sigmoid$, with examples of the effect of different values of $steepness$, are shown in Figure 6.7. in Section 6.3.3, we experiment with different values of $steepness$ and how that affects the performance of our model.

6.3.3 Results

Figure 6.8 shows our model's performance with different $min\_samples$ values, and the associated multiplier n, which affects the minimum number of opinions collected (ρ) to perform the clustering. Recall that $\rho = n.min\_samples$, so a greater value of n reflects a bigger time window size. From Figure 6.8, we note that the best model performance is for $min\_samples = 3$, when $n = 1$. We can also see that the model's overall performance

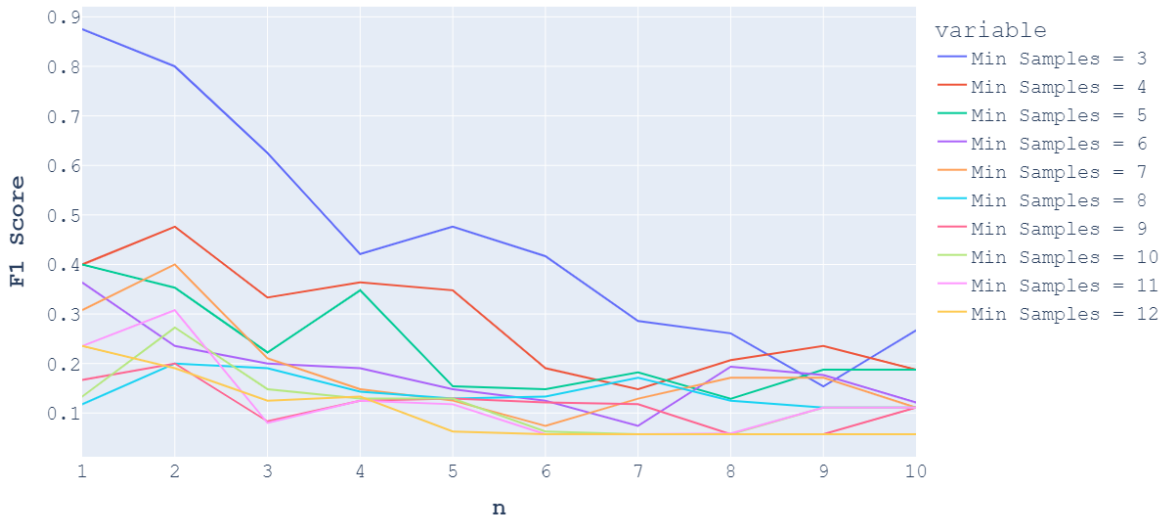

Figure 6.8: Model performance as a function of the $min\_samples$ multiplier n, with different values of $min\_samples$

decreases as $min\_samples$ increases, and as n increases as well. So, as a best performance metric, we choose $min\_samples = 3$, but we choose $n = 2$ and not 1, since we note an increase in the overall performance of the model from $n = 1$ to $n = 2$ for most of the values of $min\_samples$. We argue that this provides better generalization for different datasets.

After selecting $min\_samples$ and n, which dictate the time window size, we use them to tune for ϵ, which is determined by the standard deviation of the sentiments within a determined time window, and the $steepness$. Figure 6.9 shows the model's performance with respect to the $steepness$ parameter. Recall that for a given standard deviation, a smaller value of $steepness$ results in a larger value of ϵ, making it more likely for a document of uncommon sentiments, which are likely to be documents of fake authorship, to be undetected as anomalies, causing more false negatives. On the other hand, too large a value of $steepness$ results in a very small value of ϵ, causing more false positives. And Figure 6.9 shows an ideal value of $steepness$ for the provided dataset to be between 12 and 15, where the model's F1 score is 0.875.

6.4 Conclusion

Our findings show that the inclusion of the temporal aspect, when it comes to author verification, plays a significant role on performance. Starting with the ideal scenario, through

Figure 6.9: Model performance as a function of *steepness*

the conciseness and abundance of opinions, we were able to show that not including the temporal aspect shows a significant drop in performance, due to the fact that related works show that authors tend to change their opinions over time. Then, tuning the clustering algorithm by considering the density of opinions and their spread of polarity (standard deviation), we were able to produce high performance in determining documents of true and false authorship on a sub-dataset of our authors' twitter dataset.

It should be noted that further experimentation needs to be applied to be able to properly evaluate the model, with more authors, data per author and topics. However, based on our findings, we argue that not including the temporal aspect would negatively affect the performance of the model, since it provides us with a more current representation on an authors' opinion over time. As mentioned before, data collected for this task does not have to be restricted to 1 source, like in our case where we've only collected information from Twitter.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Having trust is a cornerstone for any online activity. Any form of online interaction requires information sharing, and not having trust in the information received, or with the destination to which the information is being sent is a greatly hindering factor in advancing and improving the online experience. This is especially true with the use of social media, where sharing personal information, and accepting any piece of information as truth without validation, is quite common. The 2 parties with regards to online information sharing are: the users, who are the people that use social networks, and share their own personal and non-personal information, and the data holders, who collect user information available on said social networks. It greatly benefits both parties to improve the information-sharing process, by adding security features to preserve respect to users privacy, and provide means for validating the authenticity of information being shared online. From here, we addressed 2 critical topics in this thesis: user privacy preservation in textual data, and author verification in micro-messages.

In Chapter 1, we provided a comprehensive introduction to textual data privacy and author verification, and the difficulty of working with text, and with micro-messages in particular. We provide the motivation behind our work, and list the different challenges that the tackled topics provide. Furthermore, the goal and objectives were described and contributions briefly explained in this Chapter.

Chapter 2 presented a review of existing solutions for privacy preservation in both structured and unstructured (textual) data, and those for author verification and how they lack the ability to handle micro-messages. The review was focused on the strength and weak-

nesses of the state of the art. and this paved the way for determining the shortcomings in the fields of privacy and AV.

In Chapter 3, we presented a theoretical approach for privacy preservation in textual data. We introduce the concept of a document's criticality, which is determined by how sensitive the information present inside of it are, and is it possible to identify the document's owner. This is done by extracting and ranking the document's identifying and sensitive terms according to their sensitivity and identifiability, and providing metric values to these rankings. Finally, if a document meets a certain threshold of sensitivity and identifyability, the appropriate anonymization procedure can be taken.

In Chapter 4, we focused on the problem of Author Verification in micro-messages. We introduced a novel semantic-based author verification approach, and used twitter tweets as our dataset. We experimented with one of the top performing algorithms of the PAN 15 AV task, and show that it performs poorly when handling micro-messages. Also, we experiment with our sentiment-basedauthor verifier on the tweets dataset, and compared the results and performance.

In Chapter 5, we improved our semantic-based author verifier from Chapter 4 by applying a clustering technique to the sentiments for the verification of true authorship, and introduced the different manners in which we tune the hyper parameters. We also collected a new micro-messages dataset of tweets published by the 88 most followed twitter influencers. We also tested the state of the art for author verification in short text, Adominem, and compared its performance on our Twitter influencers dataset. Finally, we showed that our approach is significantly more resistant than the state of the art to impersonation attacks, by crafting a set of manipulated tweets, where we alter the opinion without affecting the writing style.

Finally, in Chapter 6, we investigated the effect of the temporal aspect of data on author verification, and analyzed the extent of authors'opinion variation per topic over time, and the consequences of such variation on opinion-based author verification. We developed an artificial dataset of opinions, and then modeled the temporal effect problem as a classification problem, extended from our work in Chapter 5. Finally, we applied it to our dataset of influencers from Chapter 5.

7.2 Future Work

Our work, discussions and results presented in thesis provide a number of inviting avenues of future work. We discuss it with regards to textual privacy and author verification in micro-messages below:

Future Work in Textual Privacy

We provided a theoretical approach for textual privacy, so the main focus of our future work will be applying our approach to real-life data. This comes with 3 potential improvements:

1. Instead of relying just on domain experts to rank sensitive information in terms of their sensitivity, a data-oriented ranking method would be an improvement. One interesting direction we can address this is by analyzing people's willingness to share particular categories of information, and give the highest sensitivity rank to the category with less likelihood to be shared.

2. For a selected set of business domains, we can run an analysis of these businesses' domains to generate their associated seeds $SEED_I$ and $SEED_S$. This would help to provide a standard for future extensions into new domains.

3. With the aim of producing texts with higher analytical potential after anonymization, instead of masking the identifiers in the anonymization process, we can use a generalization approach, where each identifier is replaced with a more generic term. For example, an age would be replaced with a range of ages, and an address would be replaced with a city na,e or country name. The latter point of generalization could be achieved by using word embeddings, since it is known that by creating a translation vector, say from the word embedding of Paris to the word embedding of France, it can be used to generalize other cities like Berlin to Germany. The same concept might be applied to generalize different identifiers.

Future Work in Author Verification

Our work in author verification could pave the way for a new mentality for tackling AV problems by putting more focus on semantics. We see a number of ways where we can improve the results of our work and expand upon them:

1. The use of aspect-based sentiment analysis could provide higher accuracy in estimating the author's opinion about a text's topic, rather than using the entire text's sentiment.

2. Combining our semantic-based approach with a stylistic approach could make a great improvement on the results of our work.

3. Further experimentation related to the temporal aspect is needed, with more authors, data per author and topics.

Bibliography

[1] Ahmed Abbasi and Hsinchun Chen. "Writeprints: A stylometric approach to identity-level identification and similarity detection in cyberspace". In: *ACM Trans. Inf. Syst.* 26.2 (2008), 7:1–7:29.

[2] Anita Allen. "Privacy and Medicine". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Spring 2021. Metaphysics Research Lab, Stanford University, 2021.

[3] Eiman Alothali et al. "Detecting Social Bots on Twitter: A Literature Review". In: *2018 International Conference on Innovations in Information Technology (IIT)*. 2018, pp. 175–180.

[4] *Apple Adopts Differential Privacy*. https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf. Accessed: 2019-09-18.

[5] Eiji Aramaki et al. "Automatic deidentification by using sentence features and label consistency". In: *i2b2 Workshop on Challenges in Natural Language Processing for Clinical Data* (Jan. 2006).

[6] Maryam Archie et al. "Who's Watching ? De-anonymization of Netflix Reviews using Amazon Reviews". In: 2018.

[7] Erman Ayday et al. "Protecting and Evaluating Genomic Privacy in Medical Tests and Personalized Medicine". In: *Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society*. WPES '13. Berlin, Germany: Association for Computing Machinery, 2013, 95–106. ISBN: 9781450324854.

[8] Bruce Beckwith et al. "Development and evaluation of an open source software tool for deidentification of pathology reports". In: *BMC Medical Informatics Decis. Mak.* 6 (2006), p. 12.

[9] Jules J. Berman. "Concept-Match Medical Data Scrubbing". In: *Archives of Pathology & Laboratory Medicine* 127.6 (2003), 680–686.

[10] Janek Bevendorff et al. "Bias Analysis and Mitigation in the Evaluation of Authorship Verification". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, July 2019, pp. 6301–6306.

[11] Benedikt T. Boenninghoff et al. "Deep Bayes Factor Scoring for Authorship Verification". In: *ArXiv* abs/2008.10105 (2020).

[12] Olivier Bousquet and André Elisseeff. "Stability and Generalization". In: *Journal of Machine Learning Research* 2 (June 2002), pp. 499–526.

[13] Zhiqi Bu et al. "Deep Learning with Gaussian Differential Privacy". In: *Harvard Data Science Review* (July 2020).

[14] P. Buddha Reddy et al. "A Novel Approach for Authorship Verification". In: *Data Engineering and Communication Technology*. Springer Singapore, 2020, pp. 441–448.

[15] Ulrich Burgbacher and Klaus Hinrichs. "An Implicit Author Verification System for Text Messages Based on Gesture Typing Biometrics". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: Association for Computing Machinery, 2014, 2951–2954. ISBN: 9781450324731.

[16] Daniel Castro Castro et al. "Authorship Verification, Average Similarity Analysis". In: *Proceedings of the International Conference Recent Advances in Natural Language Processing*. INCOMA Ltd. Shoumen, BULGARIA, Sept. 2015, pp. 84–90.

[17] P. Comon. "Independent component analysis, A new concept?" In: *Signal Process.* 36 (1994), pp. 287–314.

[18] Keith Cortis et al. "SemEval-2017 Task 5: Fine-Grained Sentiment Analysis on Financial Microblogs and News". In: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, 2017.

[19] Michiel van Dam and Claudia Hauff. "Large-Scale Author Verification: Temporal and Topical Influences". In: *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*. 2014, 1039–1042.

[20] Clayton Allen Davis et al. "BotOrNot". In: *Proceedings of the 25th International Conference Companion on World Wide Web - WWW '16 Companion*. ACM Press, 2016.

[21] *De-identification and Heart Disease Risk Factors Challenge*. https://portal.dbmi.hms.harvard.edu/projects/n2c2-2014/. Accessed: 2019-10-21.

[22] *Deidentification and Smoking Challenge*. https://portal.dbmi.hms.harvard.edu/projects/n2c2-2006/. Accessed: 2019-10-20.

[23] John P. Dickerson, Vadim Kagan, and V.S. Subrahmanian. "Using sentiment to detect bots on Twitter: Are humans more opinionated than bots?" In: *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*. 2014, pp. 620–627.

[24] M. Douglass et al. "Computer-assisted de-identification of free text in the MIMIC II database". In: *Computers in Cardiology, 2004*. 2004, pp. 341–344.

[25] M.M. Douglass et al. "De-identification algorithm for free-text nursing notes". In: Feb. 2005, pp. 331 –334. ISBN: 0-7803-9337-6.

[26] Cynthia Dwork and Aaron Roth. "The Algorithmic Foundations of Differential Privacy". In: *Found. Trends Theor. Comput. Sci.* 9.3-4 (2014), pp. 211–407.

[27] *Elon Musk said use Signal, and confused investors sent the wrong stock up 438% on Monday*. https://www.cnbc.com/2021/01/11/signal-advance-jumps-another-438percent-after-elon-musk-fueled-buying-frenzy.html. Accessed: 2021-04-15.

[28] *Enabling developers and organizations to use differential privacy*. https://developers.googleblog.com/2019/09/enabling-developers-and-organizations.html. Accessed: 2019-10-20.

[29] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: AAAI Press, 1996, pp. 226–231.

[30] Natasha Fernandes, Mark Dras, and Annabelle McIver. "Generalised Differential Privacy for Text Document Processing". In: *Principles of Security and Trust*. Ed. by Flemming Nielson and David Sands. Cham: Springer International Publishing, 2019, pp. 123–148. ISBN: 978-3-030-17138-4.

[31] Natasha Fernandes, Mark Dras, and Annabelle McIver. *Generalised Differential Privacy for Text Document Processing*. 2019. arXiv: 1811.10256 [cs.CR].

[32] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. "Incorporating non-local information into information extraction systems by Gibbs sampling". In: *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics - ACL '05*. Association for Computational Linguistics, 2005.

[33] F. Jeff Friedlin and Clement J. McDonald. "Application of Information Technology: A Software Tool for Removing Patient Identifying Information from Clinical Documents". In: *J. Am. Medical Informatics Assoc.* 15.5 (2008), pp. 601–610.

[34] Karl Pearson F.R.S. "LIII. On lines and planes of closest fit to systems of points in space". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), pp. 559–572.

[35] Matthew Gabel and Kenneth Scheve. "Estimating the Effect of Elite Communications on Public Opinion Using Instrumental Variables". In: *American Journal of Political Science* 51.4 (Oct. 2007), pp. 1013–1028.

[36] Axel Gelfert. "Fake News: A Definition". In: *Informal Logic* 38.1 (2018), 84–117.

[37] *Global digital population as of January 2021 (in billions)*. https://www.statista.com/statistics/617136/digital-population-worldwide/. Accessed: 2021-08-12.

[38] Roberto J. González. "Hacking the citizenry?: Personality profiling, 'big data' and the election of Donald Trump". In: *Anthropology Today* 33.3 (2017), 9–12.

[39] *Google's Sundar Pichai was grilled on privacy, data collection, and China during congressional hearing*. https://www.cnbc.com/2018/12/11/google-ceo-sundar-pichai-testifies-before-congress-on-bias-privacy.html. Accessed: 2019-10-20.

[40] Philip John Gorinski et al. *Named Entity Recognition for Electronic Health Records: A Comparison of Rule-based and Machine Learning Approaches*. 2019. arXiv: 1903.03985 [cs.CL].

[41] Archana Goyal, Vishal Gupta, and Manish Kumar. "Recent Named Entity Recognition and Classification techniques: A systematic review". In: *Computer Science Review* 29 (Aug. 2018), pp. 21–43.

[42] Maarten Grootendorst. *KeyBERT: Minimal keyword extraction with BERT.* Version v0.1.3. 2020.

[43] Y. Guo et al. "Identifying Personal Health Information Using Support Vector Machines". In: 2006.

[44] Dilip Gupta, Melissa Saul, and John Gilbertson. "Evaluation of a Deidentification (De-Id) Software Engine to Share Pathology Reports and Clinical Documents for Research". In: *American Journal of Clinical Pathology* 121.2 (2004), 176–186.

[45] Oren Halvani, Lukas Graner, and Roey Regev. "TAVeer: An Interpretable Topic-Agnostic Authorship Verification Method". In: *Proceedings of the 15th International Conference on Availability, Reliability and Security*. New York, NY, USA: Association for Computing Machinery, 2020.

[46] Tanjim Ul Haque, Nudrat Nawal Saber, and Faisal Muhammad Shah. "Sentiment analysis on large scale Amazon product reviews". In: *2018 IEEE International Conference on Innovative Research and Development (ICIRD)*. 2018, pp. 1–6.

[47] Zellig S. Harris. "Distributional Structure". In: 10.2-3 (Aug. 1954), pp. 146–162.

[48] Ruining He and Julian J. McAuley. "Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering". In: *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016*. Ed. by Jacqueline Bourdeau et al. ACM, 2016, pp. 507–517.

[49] DAVID I. HOLMES. "The Evolution of Stylometry in Humanities Scholarship". In: *Literary and Linguistic Computing* 13.3 (Sept. 1998), pp. 111–117. ISSN: 0268-1145.

[50] Matthew Honnibal et al. *spaCy: Industrial-strength Natural Language Processing in Python*. 2020.

[51] JF Hoorn et al. "Neural network identification of poets using letter sequences". In: *Literary and Linguistic Computing* 14.3 (Sept. 1999), pp. 311–338. ISSN: 0268-1145.

[52] Clayton J. Hutto and Eric Gilbert. "VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text". In: *Proceedings of the Eighth ICWSM, June 1-4, 2014*. The AAAI Press, 2014.

[53] *If you have a smart TV, take a closer look at your privacy settings*. https://www.cnbc.com/2017/03/09/if-you-have-a-smart-tv-take-a-closer-look-at-your-privacy-settings.html. Accessed: 2019-10-12.

[54] Janek et al. "Overview of PAN 2020: Authorship Verification, Celebrity Profiling, Profiling Fake News Spreaders on Twitter, and Style Change Detection". In: Sept. 2020, pp. 372–383.

[55] Patrick Juola and Efstathios Stamatatos. "Overview of the author identification task at PAN 2013". In: *CEUR Workshop Proceedings* 1179 (Jan. 2013).

[56] Mehmet Kayaalp et al. "De-identification of Address, Date, and Alphanumeric Identifiers in Narrative Clinical Reports". In: *AMIA 2014, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 15-19, 2014*. AMIA, 2014. URL: https://knowledge.amia.org/56638-amia-1.1540970/t-004-1.1544972/f-004-1.1544973/a-174-1.1545151/a-175-1.1545148.

[57] Mehmet Kayaalp et al. "De-identification of Address, Date, and Alphanumeric Identifiers in Narrative Clinical Reports". In: *AMIA 2014, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 15-19, 2014*. AMIA, 2014. URL: https://knowledge.amia.org/56638-amia-1.1540970/t-004-1.1544972/f-004-1.1544973/a-174-1.1545151/a-175-1.1545148.

[58] Daniel Kifer and Ashwin Machanavajjhala. "No free lunch in data privacy". In: *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2011, Athens, Greece, June 12-16, 2011*. Ed. by Timos K. Sellis et al. ACM, 2011, pp. 193–204.

[59] Youngjun Kim, Paul M. Heider, and Stéphane M. Meystre. "Ensemble-based Methods to Improve De-identification of Electronic Health Record Narratives". In: *AMIA 2018, American Medical Informatics Association Annual Symposium, San Francisco, CA, November 3-7, 2018*. AMIA, 2018. URL: https://knowledge.amia.org/67852-amia-1.4259402/t004-1.4263758/t004-1.4263759/2976309-1.4263922/2975300-1.4263919.

[60] Mirco Kocher and Jacques Savoy. "UniNE at CLEF 2015 Author Identification: Notebook for PAN at CLEF 2015". In: *Working Notes of CLEF 2015 - Conference and Labs of the Evaluation forum, Toulouse, France, September 8-11, 2015*. Vol. 1391. CEUR Workshop Proceedings. CEUR-WS.org, 2015.

[61] Moshe Koppel, Jonathan Schler, and Shlomo Argamon. "Authorship attribution in the wild". In: *Lang. Resour. Evaluation* 45.1 (2011), pp. 83–94.

[62] Moshe Koppel, Jonathan Schler, and Kfir Zigdon. "Determining an author's native language by mining a text for errors". In: *Proceedings of the Eleventh ACM SIGKDD 21-24, 2005*. ACM, 2005, pp. 624–628.

[63] Moshe Koppel and Yaron Winter. "Determining if two documents are written by the same author". In: *J. Assoc. Inf. Sci. Technol.* 65.1 (2014), pp. 178–187.

[64] Moshe Koppel et al. "The "Fundamental Problem" of Authorship Attribution". In: *English Studies* 93 (May 2012), pp. 284–291.

[65] Akshi Kumar et al. "Hybrid context enriched deep learning model for fine-grained sentiment analysis in textual and visual semiotic modality social data". In: *Information Processing & Management* 57.1 (Jan. 2020), p. 102141.

[66] Zhenzhong Lan et al. "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations". In: *arXiv:1909.11942 [cs]* (2020). arXiv: 1909.11942. URL: http://arxiv.org/abs/1909.11942.

[67] David M. J. Lazer et al. "The science of fake news". In: *Science* 359.6380 (2018), pp. 1094–1096. ISSN: 0036-8075. eprint: https://science.sciencemag.org/content/359/6380/1094.full.pdf.

[68] Ji Young Lee et al. "Feature-Augmented Neural Networks for Patient Note De-identification". In: *Proceedings of the Clinical Natural Language Processing Workshop, ClinicalNLP@COLING 2016, Osaka, Japan, December 11, 2016*. Ed. by Anna Rumshisky et al. The COLING 2016 Organizing Committee, 2016, pp. 17–22. URL: https://aclanthology.org/W16-4204/.

[69] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. "t-Closeness: Privacy Beyond k-Anonymity and l-Diversity". In: *Proceedings of the 23rd International Conference on Data Engineering, ICDE 2007, The Marmara Hotel, Istanbul, Turkey, April 15-20, 2007*. Ed. by Rada Chirkova et al. IEEE Computer Society, 2007, pp. 106–115.

[70] Bing Liu. *Sentiment Analysis*. Cambridge University Press, 2015.

[71] Ashwin Machanavajjhala et al. "l-Diversity: Privacy Beyond k-Anonymity". In: *Proceedings of the 22nd International Conference on Data Engineering, ICDE 2006, 3-8 April 2006, Atlanta, GA, USA*. Ed. by Ling Liu et al. IEEE Computer Society, 2006, p. 24.

[72] David Maier. *The Theory of Relational Databases*. Computer Science Press, 1983. ISBN: 0-914894-42-0. URL: http://web.cecs.pdx.edu/\%7Emaier/TheoryBook/TRD.html.

[73] Michał Marcińczuk, Jan Kocoń, and Michał Gawor. "Recognition of Named Entities for Polish-Comparison of Deep Learning and Conditional Random Fields Approaches". In: *Proceedings of the PolEval 2018 Workshop*. Ed. by Maciej Ogrodniczuk and Łukasz Kobyliński. Warsaw, Poland: Institute of Computer Science, Polish Academy of Science, 2018, pp. 77–92.

[74] Julian J. McAuley et al. "Image-Based Recommendations on Styles and Substitutes". In: *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*. Ed. by Ricardo Baeza-Yates et al. ACM, 2015, pp. 43–52.

[75] Tomas Mikolov et al. "Distributed Representations of Words and Phrases and their Compositionality". In: *Advances in Neural Information Processing Systems* 26 (Oct. 2013).

[76] Jennifer C. Molloy. "The Open Knowledge Foundation: Open Data Means Better Science". In: *PLoS Biology* 9.12 (Dec. 2011), e1001195.

[77] Arvind Narayanan and Vitaly Shmatikov. "Robust De-anonymization of Large Sparse Datasets". In: *2008 IEEE Symposium on Security and Privacy (S&P 2008), 18-21 May 2008, Oakland, California, USA*. IEEE Computer Society, 2008, pp. 111–125.

[78] Richard J. Oentaryo et al. "On Profiling Bots in Social Media". In: *Social Informatics*. Ed. by Emma Spiro and Yong-Yeol Ahn. Cham: Springer International Publishing, 2016, pp. 92–109.

[79] Eleni Partalidou et al. "Design and implementation of an open source Greek POS Tagger and Entity Recognizer using spaCy". In: *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. 2019, pp. 337–341.

[80] Anselmo Peñas and Álvaro Rodrigo. "A Simple Measure to Assess Non-response". In: *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24*. The Association for Computer Linguistics, 2011, pp. 1415–1424.

[81] Fuchun Peng, Dale Schuurmans, and Shaojun Wang. "Augmenting Naive Bayes Classifiers with Statistical Language Models". In: *Inf. Retr.* 7.3-4 (2004), pp. 317–345.

[82] Kashif Riaz. "Rule-Based Named Entity Recognition in Urdu". In: *Proceedings of the 2010 Named Entities Workshop*. NEWS '10. Uppsala, Sweden: Association for Computational Linguistics, 2010, 126–135. ISBN: 9781932432787.

[83] Anderson Rocha et al. "Authorship Attribution for Social Media Forensics". In: *IEEE Trans. Inf. Forensics Secur.* 12.1 (2017), pp. 5–33.

[84] Pierangela Samarati and Latanya Sweeney. *Protecting Privacy when Disclosing Information: k-Anonymity and Its Enforcement through Generalization and Suppression.* Tech. rep. 1998.

[85] Roy Schwartz et al. "Authorship Attribution of Micro-Messages". In: *Proceedings of the 2013 Conference on EMNLP*. ACL, 2013, pp. 1880–1891.

[86] Xingjian Shi et al. "Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting". In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*. Ed. by Corinna Cortes et al. 2015, pp. 802–810. URL: https://proceedings.neurips.cc/paper/2015/hash/07563a3fe3bbe7e3ba84431ad9d055af-Abstract.html.

[87] Prasha Shrestha et al. "Convolutional Neural Networks for Authorship Attribution of Short Texts". In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*. Association for Computational Linguistics, 2017, pp. 669–674.

[88] Kai Shu et al. "Fake News Detection on Social Media: A Data Mining Perspective". In: *SIGKDD Explor. Newsl.* 19.1 (Sept. 2017), 22–36. ISSN: 1931-0145.

[89] Kai Shu et al. "The Role of User Profiles for Fake News Detection". In: *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. ASONAM '19. Vancouver, British Columbia, Canada: Association for Computing Machinery, 2019, 436–439. ISBN: 9781450368681.

[90] Jaspreet Singh, Gurvinder Singh, and Rajinder Singh. "A review of sentiment analysis techniques for opinionated web text". In: *CSI Transactions on ICT* 4.2-4 (Dec. 2016), pp. 241–247.

[91] Zeenia Singla, Sukhchandan Randhawa, and Sushma Jain. "Statistical and sentiment analysis of consumer product reviews". In: *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. 2017, pp. 1–6.

[92] Thamar Solorio et al. "Modality Specific Meta Features for Authorship Attribution in Web Forum Posts". In: *Fifth IJCNLP*. 2011, pp. 156–164.

[93] Yu Song et al. "POSBIOTM—NER: a trainable biomedical named-entity recognition system". In: *Bioinformatics* 21.11 (Apr. 2005), pp. 2794–2796. ISSN: 1367-4803. eprint: https://academic.oup.com/bioinformatics/article-pdf/21/11/2794/834139/bti414.pdf.

[94] Efstathios Stamatatos. "Authorship Verification: A Review of Recent Advances". In: *Research in Computing Science* 123 (Dec. 2016), pp. 9–25.

[95] Efstathios Stamatatos et al. “Overview of the author identification task at PAN 2014”. In: *CEUR Workshop Proceedings* 1180 (Jan. 2014), pp. 877–897.

[96] Efstathios Stamatatos et al. “Overview of the Author Identification Task at PAN 2015”. In: Sept. 2015.

[97] *Standards for Privacy of Individually Identifiable Health Information.* URL: https://www.federalregister.gov/documents/2002/08/14/02-20554/standards-for-privacy-of-individually-identifiable-health-information.

[98] Chanchal Suman et al. “Emoji Helps! A Multi-modal Siamese Architecture for Tweet User Verification”. In: *Cognitive Computation* 13.2 (2020), 261–276.

[99] Latanya Sweeney. *Simple Demographics Often Identify People Uniquely.* 2018. URL: https://kilthub.cmu.edu/articles/journal_contribution/Simple_Demographics_Often_Identify_People_Uniquely/6625769/1.

[100] György Szarvas, Richárd Farkas, and András Kocsor. “A Multilingual Named Entity Recognition System Using Boosting and C4.5 Decision Tree Learning Algorithms”. In: *Discovery Science, 9th International Conference, DS 2006, Barcelona, Spain, October 7-10, 2006, Proceedings.* Ed. by Ljupco Todorovski, Nada Lavrac, and Klaus P. Jantke. Vol. 4265. Lecture Notes in Computer Science. Springer, 2006, pp. 267–278.

[101] *The Facebook and Cambridge Analytica scandal, explained with a simple diagram.* https://www.vox.com/policy-and-politics/2018/3/23/17151916/facebook-cambridge-analytica-trump-diagram. Accessed: 2019-10-20.

[102] Rob Trent. *Elon Musk: Driving Cars Will Be Banned.* Youtube. 2015. URL: https://www.youtube.com/watch?v=Q6q-HZBLy0I&t=21s.

[103] *Trump order bans US firms from dealing with Huawei.* https://www.techradar.com/news/trump-order-bans-us-firms-from-dealing-with-huawei. Accessed: 2019-10-25.

[104] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems 30.* 2017, pp. 5998–6008.

[105] Gillian Warner-Søderholm et al. “Who trusts social media?” In: *Computers in Human Behavior* 81 (2018), pp. 303–315. ISSN: 0747-5632.

[106] Benjamin Weggenmann and Florian Kerschbaum. “SynTF: Synthetic and Differentially Private Term Frequency Vectors for Privacy-Preserving Text Mining”. In: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018.* Ed. by Kevyn Collins-Thompson et al. ACM, 2018, pp. 305–314.

[107] Ben Wellner et al. “Research Paper: Rapidly Retargetable Approaches to De-identification in Medical Records”. In: *J. Am. Medical Informatics Assoc.* 14.5 (2007), pp. 564–573.

[108] Miguel Won, Patricia Murrieta-Flores, and Bruno Martins. "Ensemble Named Entity Recognition (NER): Evaluating NER Tools in the Identification of Place Names in Historical Corpora". In: *Frontiers in Digital Humanities* 5 (Mar. 2018).

[109] Vikas Yadav and Steven Bethard. *A Survey on Recent Advances in Named Entity Recognition from Deep Learning models*. 2019. arXiv: 1910.11470 [cs.CL].