Author Identity Unveiled: Gender and Age Prediction from Textual Patterns using BERT

Balla Charishma Sulochana, Bhavya Sri Pragada, Boya Chaitanya Kiran, Gaddam Anvith Reddy,
Manju Venugopalan*
Department of Computer Science and Engineering,
Amrita School of Computing, Bengaluru,
Amrita Vishwa Vidyapeetham, India
charisma.sulochana@gmail.com, bhavyasri.pragada@gmail.com, chaitanyakiran415@gmail.com,
ganvithreddy3052@gmail.com, v_manju@blr.amrita.edu*

***Abstract*- The research delves into author profiling, aiming to identify writers' age groups and genders using extensive textual data. This involves utilizing BERT embeddings to understand sentence structure, word selection, and the overall tone of the writing. The primary goal is to employ supervised machine learning models to categorize authors based on age and gender, aiding marketers in tailoring campaigns for diverse demographics. As for text vectorization methods, BERT embeddings stand out for their ability to capture contextual information efficiently. The objective of this research is to develop a robust solution for author profiling. Additionally, techniques like TF-IDF is also considered effective for text representation, each offering unique advantages depending on the specific task and dataset characteristics. The outcome of this study strongly indicates that the AdaBoost with TF-IDF stands out as the most precise machine learning models for age group prediction with an F1 score of 0.58. The K-neighbours with TF-IDF is the best model for gender prediction with an F1 score of 0.58.**

Keywords: BERT, Author Profiling, Age, Gender, Machine Learning, Feature Extraction, TF-IDF.

I. INTRODUCTION

Author profiling, a fascinating field within natural language processing, revolves around extracting demographic information, particularly age and gender, from written text. By delving into the linguistic fine points of an author's expressions [1], machine learning models can discern patterns that correlate with specific age groups and genders. Author profiling involves studying written text to infer information about the author, such as age and gender, by analysing linguistic patterns.

Some key steps are used in making author profiles to get important data and features. One of the main method [2] is taking out language features. This means looking at things like words used, sentence structures and writing styles in a way that we can count or measure them. These points make a picture that can show facts about the writer's age, character or culture. Also, feeling analysis is very important because the emotional mood of what's written can give hints about how the writer feels and views things. In today's digital age, this practice is highly useful as it enables tailored approaches in various areas. For instance, businesses can refine marketing strategies based on the preferences of different age groups and genders. Educational systems can customize learning materials, and forensic investigations can benefit from understanding writing styles.

Machine learning methods, especially supervised learning systems, are often used to create forecast models from labelled training data [3]. These models can then be used to put new texts into categories and guess who wrote them. Social network study is a keyway, looking at the links between writers and others in online groups to find patterns and connections. In the end, a mix of language study, computer learning. and social network strategies make up strong tools for writer identifying. This lets experts understand more about who's writing these texts. Natural language processing (NLP) is important for understanding the writer's profile. It helps to decode written words and figure out things like age or gender from them.

In this work, understanding words and speech (NLP) is very important. This helps to find what to study in things which can be read like blog posts or social media content. By using NLP, Model learns what writers do. This helps create strong machine learning models for guessing ages and genders rightly. NLP works like a digital detective, finding small language clues in big databases. This helps understand writers better and makes marketing plans more effective. It also helps find real voices among lots of online content.

The diverse array of machine learning algorithms utilized in our research, showcases a comprehensive approach to solving the task of finding the characteristic insights of authors. The ensemble classifiers, including AdaBoost and Random Forest, along Logistic Regression and Linear Support Vector Machine (LinearSVC) uses a careful plan. This joins the strong parts of each algorithm to make sure their predictions are better. These models each have their own special traits and beliefs, which makes them useful for different situations when trying to find out about the

author. Group methods such as AdaBoost and Random Forest are good at finding complex links and making better guesses. Meanwhile, linear models like Logistic Regression and LinearSVC do a great job with tasks where things can be separated in straight lines. Naive Bayes models, which work with probabilities, give a way to classify things using probability. Adding a decision tree and k-nearest neighbours makes the way we model more different. Moreover, for author profiling, [4] BERT can be one of the key features of processing text. It works like a language detective, carefully looking at all parts of written stuff to find special details. It's like having a language expert look at how an author writes words. They catch small clues that tell if someone is young or old and their gender too. BERT's ability to find important details lets it see things other models might miss. It helps us understand writers better with a deeper look at their different styles and patterns in writing.

The ideology is about comparing data sets in all ways using different ML models and extractions. But its main goal is to look deeper into the patterns of authors from this information. It's important to understand and guess the ways authors write [5]. The approach integrates two distinct feature extraction methods: TF-IDF and BERT word embedding. These extractions are important in finding out who wrote something, helping to spot and group people by how old they are and if they're a guy or girl. By using TF-IDF [6] for age and gender classification, our goal was to reach the best accuracy in showing different patterns that show who authors are.

In summary, the research amalgamates machine learning models and feature extractions to decode the distinctive characteristics of authors. By leveraging SVM, Linear Regression, Random Forest, KNN, Naive Bayes, Ada Boost, TF-IDF and BERT, we aim to uncover the intricate patterns that define age and gender in textual data. Through meticulous analysis and comparison, our project contributes to the evolving landscape of author profiling, offering a nuanced understanding of linguistic patterns in the digital realm.

The following sections in the article are organized as follows. Section 2 attempts to report specific research for author profiling. While Section 3 outlines the dataset, Section 4 provides an overview of the methodology. Exploratory Data Analysis are shown in Section 5. The results and analysis are presented in Section 6, and Section 7 offers a conclusion and discussion of potential future developments.

The key contributions of the paper are:

1)Experimented traditional text vectorization methods like TF-IDF as features for input to machine learning models to handle author profiling

2)Assessed BERT embedding capabilities to enhance the results for author profiling over traditional feature extraction methods

II. RELATED WORKS

The research was made from the root of natural language processing, with the purpose of predicting specific categories through the utilization of machine learning models. The primary emphasis was placed on text vectorization and word embedding approaches, and the comparison of a number of different papers was utilized for analysis.

M Kavitha and team propose a novel method to enhance author profiling, focusing on predicting gender and age using word embeddings. Utilizing tools like XGBoost, random forests, and logistic regression, they analyze messages related to the national missing persons system on social media. Employing advanced models like Word2vec, Glove, and BERT for creating document lists improves predictive power [7]. Notably, utilizing BERT word groups in the XGBoost classifier demonstrates superior accuracy in guessing age and gender. This approach, compared to traditional methods, proves effective in understanding authors, particularly in determining gender and age.

Sarra Ouni and team [8] act as detectives in the expansive realm of author profiling, exploring diverse machine and deep learning techniques for analysing social network text. Their main goal is to craft a thorough thematic taxonomy, examining the strengths and weaknesses of various profiling solutions. The survey encompasses a range of methods, from classic Support Vector Machines and Naive Bayes to advanced ones like Artificial Neural Networks and Convolutional Neural Networks.

Paolo rosso and team [9] embark on a computational detective journey in the realm of author profiling, aiming to deduce authors' gender and age using advanced machine learning. Their analogy likens it to a sherlock holmes adventure with computers, employing a support vector machine as the analytical detective. The study emphasizes the sophistication of using writing styles for author identification and highlights the ongoing need for refining computational tools in unravelling authorial mysteries. Madhubala et al. [10] dives into author profiling using deep learning, predicting traits like age and gender from writing styles. They use FastText and glove for word embeddings, employing LSTM and CNN models on a diverse dataset of dialogues. Results show LSTM with FastText hits 60.48% accuracy for gender, and CNN with FastText scores 59.32% for age prediction. The findings highlight the power of deep learning in accurately profiling authors. Vasudeva Varma and team [11] took a machine learning approach to predict

age and gender based on writing style, achieving 64% accuracy for both age and gender in English papers, and 57% accuracy for Spanish genders. They utilized content-based features like n-grams, tags, punctuation, and topic-focused bits using latent Dirichlet allocation. The field of author profiling has been extensively studied, focusing on different methods for predicting gender and age through writing styles. Employing a machine learning technique utilizing content-based features like n-grams, tags, punctuation, and topic-specific elements have yielded impressive accuracy rates. In our study, we took this a step further by implementing text vectorizations and BERT embeddings as features.

III. DATASET

The PAN13 Age and Gender Classification Data[1], found on Zenodo.org has 200,00 documents in XML type of format. Each document is characterized by three columns: Content, which is what the document talks about; age, showing how old the writer is (10s or 20s and up); gender telling us if they are male or female. This data set can help researchers who want to make computer models for sorting things by ages and genders.

IV. PROPOSED METHODOLOGY

In this research, we are undertaking a comprehensive analysis by employing four distinct classification models, leveraging both BERT word embeddings and traditional feature extraction techniques. The methodology enables comparison between the efficacy of BERT embeddings and conventional feature extraction techniques like TF-IDF in the context of age and gender classification and is showcased in Fig.1.

A. Data Preprocessing:

The dataset is loaded into a Pandas Data Frame. It first finds and counts all the missing values in every detail columns. Then, it deletes rows with missing information to make an organized data set. Moreover, it finds and removes extra rows using the 'Content' column. In summary, this step serves as an essential data preprocessing step. It deals with missing information and repeats to make sure our data is good quality and safe before we continue doing more analysis or making models.

B. Data Visualization:

The frequency distribution of authors across distinct age groups and gender classifications in the dataset is calculated. The output consists of clear summaries of the number of authors in each age group and gender, providing valuable insights into the dataset's demographic composition.

C. Data Transformation:

Text Preprocessing employs regular expressions to remove punctuation and numeric characters, converts the text to lowercase, and eliminates excess whitespace. It begins by using regular expressions to remove HTML tags, specifically targeting patterns indicative of line breaks ("
"). This function is then applied to the 'Content' column of the Pandas Data Frame. Label encoding is another process where it assigns a numerical label to each unique category in the 'Gender' column, facilitating the use of categorical data in machine learning models. The same procedure is then repeated for the 'Age Group' column. This encoding is particularly useful when working with algorithms that require numerical inputs, enabling the incorporation of categorical features in model training and analysis.

D. Feature Extraction:

1) Text Vectorization:

Two different text vectorization techniques, TF-IDF (Term Frequency-Inverse Document Frequency) are applied to the text data. The TF-IDF vectorizer computes the TF-IDF values for each term in the document, capturing both term frequency and its importance across the entire dataset.

2) BERT:

a. BERT Tokenisation:

BERT (Bidirectional Encoder Representations from Transformers) tokenization is implemented using the Hugging Face Transformers library. The Bert Tokenizer from the 'BERT-base-uncased' pre-trained model is utilized to tokenize the processed text data in the 'Processed Text' column of the DataFrame 'df.' The encode plus method tokenizes each text sample, adds special tokens for BERT input format, ensures a consistent maximum length (in this case, 32 tokens), and returns a dictionary of tensors including the tokenized input and attention mask. The resulting tokenized representations are stored in a new column named 'Tokenized Text.' This tokenization process prepares the text data for input into BERT-based models [12], preserving the contextual relationships

[1] 10.5281/zenodo.3715863

between words and enhancing their representation for downstream natural language processing tasks.

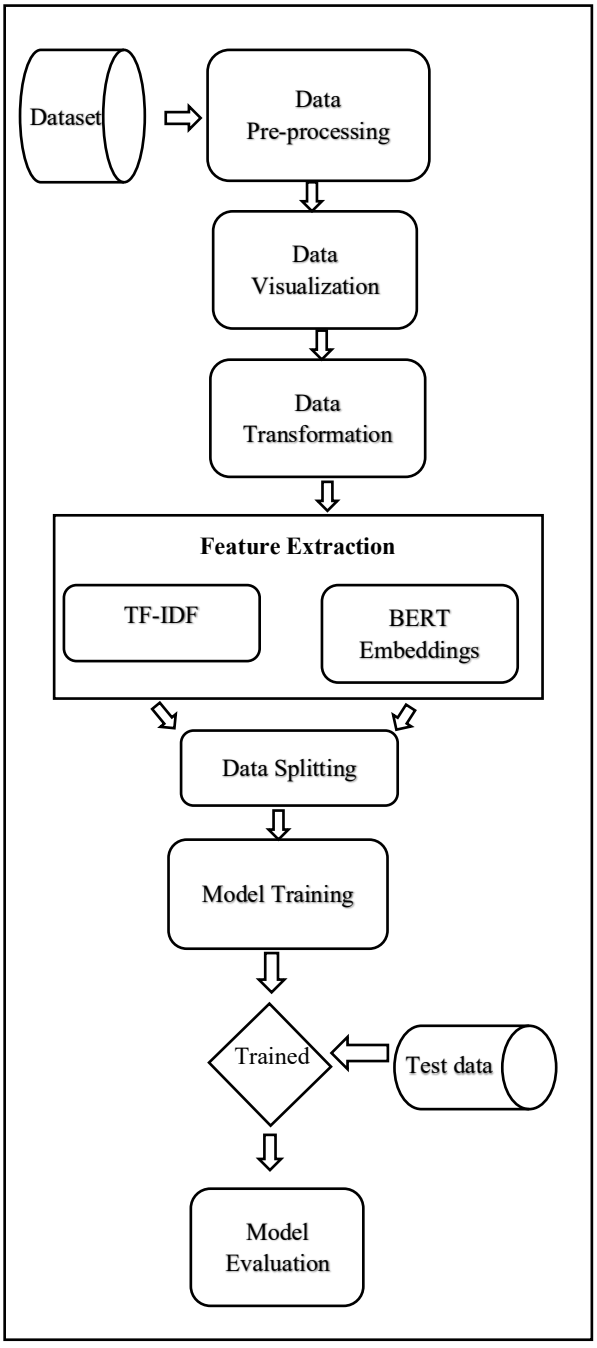

Fig. 1. Proposed methodology for author profiling

b. BERT Word embedding:

BERT embeddings [13] are generated for the tokenized text data using the 'BERT-base-uncased' pre-trained model. The function takes the tokenized input, specifically the 'input_ids,' and retrieves the BERT model's last hidden state, which contains contextualized embeddings for each token. The resulting BERT embeddings are appended to a new column named 'BERT Embeddings' in the Data Frame. This step completes the transformation of the text data into dense numerical representations, capturing contextual information for each token, and making it suitable for input into downstream machine learning models that require a numerical format.

c. Flattening and concatenate:

The BERT embeddings are flattened [14] by calculating the mean along the axis of the embeddings. The resulting flattened embeddings are then vertically stacked using NumPy to create a 2D array named 'embedding_array'. Finally, the new DataFrame is concatenated with the original one, incorporating the flattened BERT embeddings as additional features. This process transforms the BERT embeddings into a format suitable for traditional machine learning models, facilitating further analysis or model training on the enriched feature set.

Tokenized Text	BERT Embeddings	Flattened Embeddings	0	1	2	3	...	758
[input_ids, token_type_ids, attention_mask]	(((tf.Tensor(-0.31977496, shape=(), dtype=floa...	[[-0.40822378, 0.16020557, 0.10844434, 0.19822...	-0.408224	0.160206	0.108444	0.198225	...	-0.266288
[input_ids, token_type_ids, attention_mask]	(((tf.Tensor(-0.2841652, shape=(), dtype=float...	[[0.19204016, 0.20596215, 0.43610308, -0.03095...	0.192040	0.205962	0.436103	-0.030960	...	0.115800
[input_ids, token_type_ids, attention_mask]	(((tf.Tensor(0.2705576, shape=(), dtype=float3...	[[0.14549167, 0.13775417, 0.32200363, 0.113609...	0.145492	0.137754	0.322004	0.113610	...	-0.033398
[input_ids, token_type_ids, attention_mask]	(((tf.Tensor(0.34701687, shape=(), dtype=float...	[[0.2837627, 0.041847527, -0.09660588, -0.1172...	0.283763	0.041848	-0.096606	-0.117241	...	-0.077965

Fig 2: Columns added after flattening BERT embeddings

E. Data Splitting:

The dataset is split into training and testing sets. The split is configured to allocate 70% of the data to training and 30% to testing.

F. Model Training:

In the model training phase, diverse classifiers are employed, each with its unique approach to capturing patterns in the training data. In the model training section, various classifiers [15] such as Logistic Regression, K-Nearest Neighbors, Naive Bayes, Support Vector Machine (SVM), Random Forest, Decision Tree, and AdaBoost are instantiated, fitted to the training data. Once trained, the models are then applied to the test data to predict class labels. This comprehensive training process [16] lays the foundation for subsequent evaluation, enabling a comparative analysis of each model's predictive performance on the given classification task.

G. Model Evaluation:

Each model's performance metrics [17], including accuracy, precision, recall, and F1-score, are then computed and printed. This table summarizes the performance metrics for each model, offering a convenient overview for comparison. This comparison gives insights into how well each model balances correctness, completeness, and overall predictive capability. The comparison table facilitates an informed decision-making process regarding the selection of the most suitable model based on the specific requirements of the classification task at hand. Overall, this structured evaluation approach contributes to a deeper understanding of the strengths and limitations of each model in the context of the given classification problem.

V. EXPLORATORY DATA ANALYSIS

Fig.3 is generated using a pivot table to visually represent distribution of authors across various age groups and genders. By grouping the data based on 'Age Group' and 'Gender' columns, the pivot table provides an organized structure, and the resulting stacked bar chart offers a clear visualization of how authors are distributed across different demographic categories

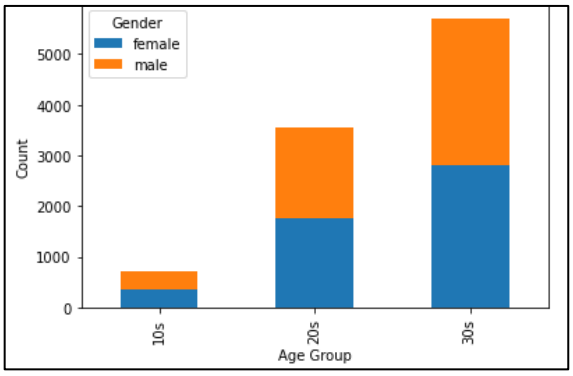

Fig 3: Bar plot representing 3 age categories

Word Clouds are created for female and male authors, respectively, within different age groups which provide a visually appealing and concise summary of textual data a few which are presented from Fig.4 to Fig.7.

Fig 4: Word cloud for age group 10's of male

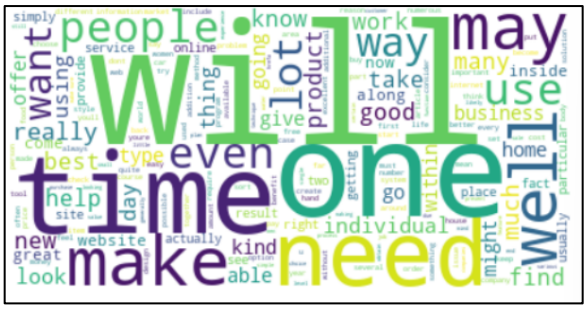

Fig 5: Word cloud for age group 30's of male

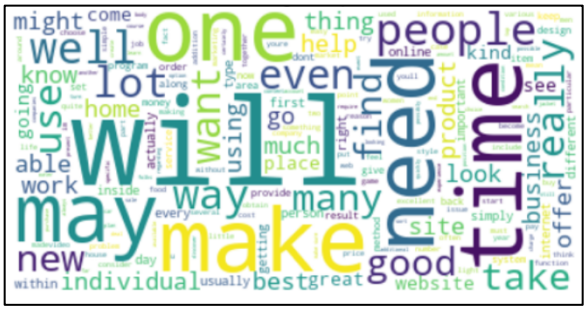

Fig 6: Word cloud for age group 10's of female

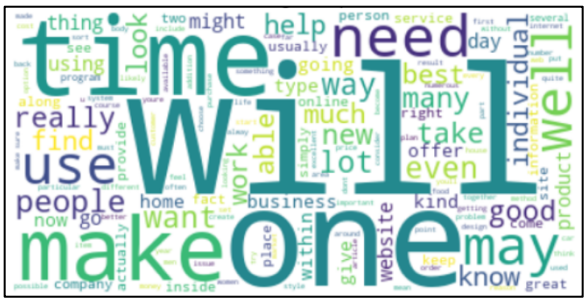

Fig 7: Word cloud for age group 30's of male

V. RESULTS AND ANALYSIS

The experiments for age group prediction using BERT is performed using multiple classifier models the results of which are reported in Table II and the best result is for Random Forest with f1 score of 0.54.

TABLE I
PERFORMANCE METRICS FOR AGE GROUP CLASSIFICATION USING BERT EMBEDDINGS

Model	Accuracy	Precision	Recall	F1
Logistic Regression	0.56	0.51	0.56	0.52
K-neighbours	0.52	0.49	0.52	0.50
Naive Bayes	0.55	0.53	0.55	0.53
SVM	0.56	0.52	0.56	0.53
Random Forest	0.60	0.56	0.60	**0.54**
Decision Tree	0.48	0.48	0.48	0.48
Ada Boost	0.58	0.52	0.58	0.53

TABLE II
PERFORMANCE METRICS FOR AGE GROUP CLASSIFICATION USING TF-IDF

Model	Accuracy	Precision	Recall	F1
Logistic Regression	0.54	0.53	0.54	0.53
K-neighbours	0.57	0.53	0.57	0.53
Naive Bayes	0.61	0.56	0.61	0.56
SVM	0.62	0.60	0.62	0.57
Random Forest	0.62	0.64	0.62	0.57
Decision Tree	0.53	0.51	0.53	0.52
Ada Boost	0.61	0.58	0.61	**0.58**

TABLE III
PERFORMANCE METRICS FOR GENDER GROUP CLASSIFICATION USING BERT EMBEDDINGS

Model	Accuracy	Precision	Recall	F1
Logistic Regression	0.53	0.54	0.53	0.53
K-neighbours	0.51	0.52	0.50	0.51
Naive Bayes	0.52	0.56	0.24	0.33
SVM	0.53	0.54	0.53	**0.54**
Random Forest	0.53	0.54	0.48	0.51
Decision Tree	0.53	0.53	0.50	0.52
Ada Boost	0.51	0.52	0.52	0.52

TABLE IV
PERFORMANCE METRICS FOR GENDER GROUP CLASSIFICATION USING TF-IDF

Model	Accuracy	Precision	Recall	F1
Logistic Regression	0.54	0.54	0.56	0.55
K-neighbours	0.53	0.53	0.63	**0.58**
Naive Bayes	0.55	0.57	0.43	0.49
SVM	0.54	0.56	0.49	0.52
Random Forest	0.55	0.57	0.46	0.51
Decision Tree	0.52	0.53	0.54	0.54
Ada Boost	0.52	0.53	0.61	0.56

The experiments are repeated for age group prediction using TF-IDF, the results of which are reported in Table II and the best result is for Ada Boost with f1 score of 0.58.Then the experiment for gender classification using BERT is performed and the results are reported in Table III and the best result is for SVM with f1 score of 0.54. The experiments are repeated for gender classification using TF-IDF and the results are reported in Table IV and the best result is for K-neighbours with f1 score of 0.58.

For age and gender classifications, both TF-IDF and BERT embeddings show competitive performance. BERT embeddings outperforming in Naive Bayes, SVM, Random Forest, Decision Tree, and Ada Boost models. Logistic Regression performs consistently well across both vectorization methods. In gender classification, TF-IDF generally yields superior results, particularly in Naive Bayes and Ada Boost models. The K-Neighbours with TF-IDF is the best model for gender prediction with an F1 score of 0.58. The outcome of this study strongly indicate that the AdaBoost with TF-IDF stands out as the most precise machine learning models for age group prediction with an F1 score of 0.58. These findings underscore the importance of considering both the vectorization method and model selection for optimal results in authorship classification tasks.

The final results of the proposed model were compared to those reported by Peiling Y and Arkaitz Zubiaga [18] where the framework consisted of four components: Concentrator, Discriminator, BERT Encoder Measurer and Small datasets adaptive classifier, whose performance metric score was 0.53, whereas our proposed models' score is 0.58. The comparison is showcased in Table V.

TABLE V
COMPARISON WITH BASELINE MODEL

Model	F-measure
Peiling et al. [18]	0.53
Proposed model	0.58

VII. CONCLUSION

In conclusion, this work makes a major impact on understanding who is the author based on the given input text. The findings suggest that the choice of vectorization method and model significantly impacts performance. The outcome of this study strongly indicates that the AdaBoost with TF-IDF stands out as the most precise machine learning models for age group prediction with an F1 score of 0.58. The K-neighbours with TF-IDF is the best model for gender

prediction with an F1 score of 0.58. BERT embeddings demonstrate notable effectiveness, particularly in complex models such as Naive Bayes, SVM, Random Forest, Decision Tree, and Ada Boost. Logistic Regression consistently performs well across vectorization methods. Adding BERT, which understands things from both directions, is helping to solve simple language patterns. It's beating old ways of using machine learning methods. It also shows practical uses in areas like customizing marketing strategies or fighting fake news by helping find real authors.

The possible future work is that there are many ways to do more study and improvement in this area. Including more complicated natural language processing (NLP) methods and deeper learning designs beyond BERT might allow us to understand better the complex details about who wrote something. The project is creating a base for future efforts to improve and grow how we study the writing of authors.

REFERENCES

[1] S. Mamgain, R. C Balabantaray and A. K Das, "Author Profiling: Prediction of Gender and Language Variety from Document," 2019 International Conference on Information Technology (ICIT), Bhubaneswar, India, 2019, pp. 473-477, doi: 10.1109/ICIT48102.2019.00089. keywords: {Predictive models;Machine learning;Data models;Natural language processing;Logistics;XML;Task analysis;NLP;Classification;Data Preprocessing;NLTK;Bag of-Wrds;LSTM CNN},

[2] S. Rathod, "Exploring Author Profiling for Fake News Detection," 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), Los Alamitos, CA, USA, 2022, pp. 1614-1619, doi: 10.1109/COMPSAC54236.2022.00256. keywords: {Measurement;Publishing;Conferences;Computational modeling;Machine learning;Writing;Feature extraction;Fake News Detection;Machine Learning;Feature Extraction;Author Profiling;Internet Security},

[3] Patra, Braja Gopal et al. "Automatic Author Profiling Based on Linguistic and Stylistic Features Notebook for PAN at CLEF 2013." Conference and Labs of the Evaluation Forum (2013).

[4] Joo, Youngjun and Inchon Hwang. "Profiling on Social Media : An Ensemble Learning Model using Various Features Notebook for PAN at CLEF 2019." (2019).

[5] E. Alzahrani, M. Al Qurashi and L. Jololian, "Comparative Analysis of the Use of Pre-Trained Models to Profile Authors' Ages and Genders," 2022 2nd International Conference on Computing and Machine Intelligence (ICMI), Istanbul, Turkey, 2022, pp. 1-7, doi: 10.1109/ICMI55296.2022.9873677.

[6] R. Bayot and T. Gonçalves, "Multilingual author profiling using word embedding averages and SVMs," 2016 10th International Conference on Software, Knowledge, Information Management & Applications (SKIMA), Chengdu, China, 2016, pp. 382-386, doi: 10.1109/SKIMA.2016.7916251.

keywords: {Blogs;Social network services;Training;Support vector machines;Dictionaries;Software;Information management;word2vec;wordembeddings;SVM;author profiling}

[7] Kavuri, K. ., & Kavitha, M. . (2023). A Word Embeddings based Approach for Author Profiling: Gender and Age Prediction. International Journal on Recent and Innovation Trends in Computing and Communication.

[8] Ouni, Sarra & Fkih, Fethi & Omri, Mohamed Nazih. (2023). A Survey of Machine Learning-based Author Profiling from Texts Analysis in Social Networks. Multimedia Tools and Applications.

[9] Francisco Rangel1,2, Paolo Rosso2. "Use of Language and Author Profiling: Identification of Gender and Age". (2013)

[10] Dr. T. Raghunadha Reddy1, B. Madhubala2, G. Varshini3, S. K. Fayaz4. "A Deep Learning Approach for Author Profiling using Word Embeddings". International Journal for Research in Applied Science & Engineering Technology (IJRASET) (2023).

[11] Santosh, Kosgi & Bansal, Romil & Shekhar, Mihir & Varma, Vasudeva. (2013). Author Profiling: Predicting Age and Gender from Blogs Notebook for PAN at CLEF 2013.

[12] A. M. Abubakar, D. Gupta and S. Palaniswamy, "Explainable Emotion Recognition from Tweets using Deep Learning and Word Embedding Models," 2022 IEEE 19th India Council International Conference (INDICON), Kochi, India, 2022, pp. 1-6.

[13] Priyanka VT, Sanjanasri J.P, Vijay Krishna Menon and Soman KP "Exploring Fake News Identification Using Word and Sentence Embeddings" accepted in the Journal of Intelligent and Fuzzy Systems, IOS Press, Netherlands (ISSN print 1064-1246, ISSN online 1875-8967). - SCIE Indexed.

[14] B. Ganesh, M, Akumar, and Dr. Soman K. P., "Amrita CEN at SemEval-2016 Task Semantic Textual Similarity: Semantic Relation from Word Embeddings in Higher Dimension", International Workshop on Semantic Evaluation (SemEval 2016). 2016

[15] G. Gressel, K., S., A, A., Thara, S., P., H., and Prabaharan Poornachandran, "Ensemble learning approach for author profiling", in Proceedings of CLEF 2014.

[16] B. Ganesh, Dr. M. Anand Kumar, and P, S. K., "Statistical Semantics in Context Space Amrita_CEN; Author Profiling", in Working Notes of CLEF 2016 - Conference and Labs of the Evaluation forum, 2016.

[17] B. VeerasekharReddy, V. N. Thatha, N. S. Biyyapu, J. S. V. G. Krishna, D. A. Sundaram and D. Sandeep, "Named Entity Recognition on Medical text by using Deep Neural Networks," 2023 4th IEEE Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2023, pp. 1-5.

[18] Peiling Yi, Arkaitz Zubiaga, "Weakly Supervised Cross-platform Teenager Detection with Adversarial BERT", HT '21, August 30–September 2, 2021, Virtual Event, Ireland.