Authorship Attribution on Kannada Text using Bi-Directional LSTM Technique

Chandrika C P, Jagadish S Kallimani
Ramaiah Institute of Technology, Bangalore-54, India
Affiliated to Visvesvaraya Technological University
Belagavi, Karnataka, India

***Abstract*—Author attribution is the field of deducing the author of an unknown textual source based on certain characteristics inherently present in the author's style of writing. Author attribution has a ton of useful applications which help automate manual tasks. The proposed model is designed to predict the authorship of the Kannada text using a sequential neural network with Bi-Directional Long Short Term Memory layers, Dense layers, Activation function and Dropout layers. Based on the nature of the data, we have used stochastic gradient descent as an optimizer that improves the learning of the proposed model. The model extracts Part of the speech tags as one of the semantic features using the N-gram technique. A Conditional random fields model is developed to assign Part of the speech tags for the Kannada text tokens, which is the base for the proposed model. The parts of the speech model achieve an overall 90% and 91% F1 score and accuracy respectively. There is no state-of-art model to compare the performance of our model with other models developed for the Kannada language. The proposed model is evaluated using the One Versus Five (1 vs 5) method and overall accuracy of 77.8% is achieved.**

Keywords*—Authorship attribution; Bi-Directional Long Short Term Memory; machine learning algorithms; parts of speech; stylometry features

I. INTRODUCTION

Authorship Attribution (AA) finds the hidden patterns in an author's writing to identify the author of an unknown text. Not much work has been done for the same, especially for the texts in the Kannada language, which is a popular Indian regional language. The authorship attribution system works primarily to predict the probability of mapping the article to its author. Authorship attribution is a field with significant applications and a long history to present a solution to the same. Recent works in this domain for foreign languages have proven to be a powerful automated tool but in Indian regional languages, the absence of a state-of-the-art method leaves scope for improvement and research. Advancements in Machine learning techniques, Natural Language Processing (NLP) and Artificial Intelligence (AI) have helped in developing a model for author attribution by learning the distinct features in the author's writing style.

Kannada, the state language of Karnataka, belongs to India, is rich in literature and culture and the Kannada-speaking people are spread around the globe. Text processing is a challenging one. Deep learning algorithms [1] like Bidirectional Encoder Representations from Transformers, a transformer based model and a hybrid model [2] composed of Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), and Gated Recurrent Unit (GRU) proved effective in text classification.

Text Classification necessitates a significant amount of time spent analyzing the contents [3]. Several parameters like large vocabulary, semantic ambiguity, and words having meaningful relationships are used to classify the text.

Now-a-days text processing in Kannada is rapidly growing, AA can be found useful in predicting the ownership of a Kannada disputed text like threat letters, suicidal notes, literature work and so on. As per our knowledge, the proposed work is a novel approach. Till now no significant work has been carried out in the Kannada language on the authorship attribution of digital text. One can find a few works which emphasize handwriting analysis [4] so there is a lot of scope in this field, especially in the local languages. There are two main approaches to authorship attribution: Profile and Instance based approaches. The former is mainly suitable for short article samples and the latter is employed for lengthy articles. The proposed model uses a profile-based approach, in which the features are extracted from short samples to create an author's profile and then trained and tested with deep learning networks. To test the owner of a Kannada handwritten document, the handwriting styles [5-6] like cursive line, font size, the thickness of line, formation of characters, spaces between characters and words, and so on are considered but when the text is digital, then different parameters have to be used for the comparisons, these parameters are referred as Stylometric features. Stylometry features are those special features used to extract a person's writing style like lexical features, which include a total number of words/sentences/ special symbols/ usage of nouns and vocabulary richness, etc. Semantic features like POS tags and content-based features etc.

In our previous works [7-8], two AA models were developed based on lexical and syntactic features using classification algorithms and the N-grams technique respectively and observed that these models predict the probability of authorship pretty well. In the proposed work, semantic features are extracted using POS tags. Deep learning techniques are popular for many NLP applications. From the survey, it is found that Deep learning networks combined with N-gram is an efficient technique for many text processing applications and it improves the performance of a model to a great extent. Bidirectional Long Short Term Memory (Bi-LSTM) is the process of constructing a neural network that can

store a sequence of information in both directions forward (future to past) and backward (past to future). Inputs run in two directions in a bidirectional LSTM, which distinguishes it from a conventional LSTM.

The process for training and testing the proposed model using BI-LSTM is shown in the diagram below. The primary objectives for developing this model are:

- To extract semantic features of an author.
- To develop a Kannada POS tagger.
- To develop the Kannada AA model using deep learning techniques.
- Performance comparison of the proposed model with other languages models.

Fig. 1 shows the process of AA, the input to this model is the cleaned labeled Kannada dataset, quality features like POS and N-grams are extracted from this, which are later trained with machine learning models like Bidirectional LSTM and during the testing phase the anonymous text is questioned and the model predicts the most suitable author for the text.

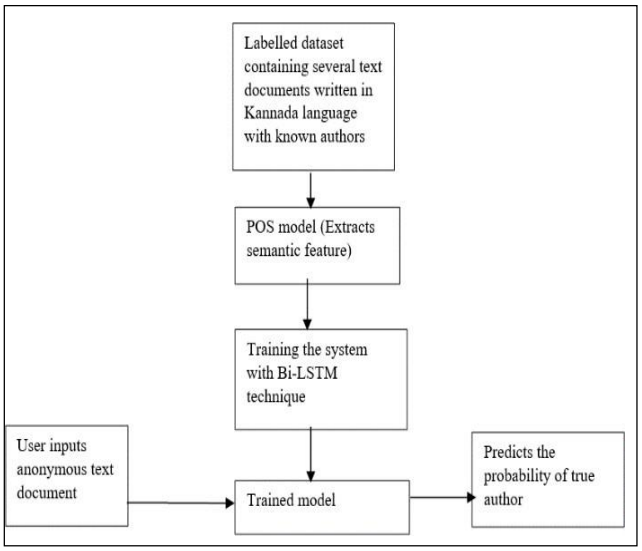

Fig. 1. Authorship Attribution Process.

Contributions

- The proposed work focuses on predicting the authorship of an anonymous Kannada text.
- The AA model accuracy mainly depends on the quality features, apart from extracting lexical features, semantic features are also extracted using the POS model, for this a POS tagger for Kannada tokens using the CRF model is developed.
- The work demonstrates the implementation of deep learning techniques like Bidirectional LSTM and using POS and grams approaches to perform AA task.
- The proposed work considers 50 authors of 500 documents and the overall accuracy of 77.8% is achieved.

II. Literature Survey

A survey will help us to analyze different techniques and methodologies explored by different researchers for Authorship attribution. This section will describe the same. A detailed survey on AA is done in [9], the authors described different dimensions of Authorship analysis including authorship prediction, verification, the importance of Stylometry features, ML algorithms and Deep learning techniques on AA. This work serves as a prerequisite for a researcher to start his work in the AA domain. Authors in [10] have explored an Instance based AA using deep learning based Artificial Neural Network on the Arabic Language. ANN (the proposed solution) produced an accuracy of 75.46% compared to 68.85%, 69.78%, 69.64%, and 69.78% attained by SVM, RF, DT and BNB respectively. Authors in [11] have explored a Cross-domain AA using Character sequences, Word uni-grams, and POS-tags features. Both the first and the second model extracts char 6-gram and 3-8-grams respectively. The third model was composed of content-based features based on POS tags. The results on the evaluation corpus are significantly lower and the three models seem to be overfitting. A different technique called Life-Like Network Automata for AA tasks is explored in [12]. This research represents network modeling texts as network automata (LLNA) with dynamics based on Life-Like rules. The LLNA method searches the whole rule space for an optimal solution to one problem. The best results were obtained with a partial lemmatization process, suggesting that this procedure is more adequate than just lemmatizing all words when text networks are used as the underlying model for this task.

Instance based and Profile-based approaches based on ensemble strategy to maximize the outcome by combining the probabilities of different feature sets by using SVM are discussed in [13]. AA task experiments on four different languages, researchers have also employed SVM with linear kernel and RBF kernel, K-nearest neighbors with K=3, and Random Forest and obtained an F1 score of 68%. Deep learning techniques proved to be very effective for AA tasks. Extracting the lexical features of an author and calculating projection for each file to predict the authorship [14] was found to be interesting. The result of the average projection shows similarities between the main author file and the summary file of each author. To recognize the true author of anonymous text written in the Russian language using deep learning networks is discussed in [15]. Authors have Extracted 33 to 5000 features and then trained and tested them using SVM and other NN features like optimizing algorithms, dropouts, loss function and various activation functions. SVM performed well with 96% average accuracy compared to a Deep neural network with 93% accuracy.

A convolution neural network based authorship model for the Bengali language is demonstrated in [16], authors have considered 6 author's 350 samples, and character level pre-trained embedding called fastText gives maximum accuracy of 98%, this work proves that pre-trained embedding outperforms compared to the non-pre-trained embedding of the text. The pre-trained models like BERT, Embeddings from Language Models(ELMo), Universal Language Model Fine-tuning and generation Generative Pre-trained Transformer -2

based authorship prediction on cross domain has been demonstrated in [17], a multi-headed classifier and DEMUX layer is created to handle different classifiers, BERT and ELMo outperform with more than 90% accuracy compared to other language models. Stylometry features play a vital role in the AA task, authors in [18] have explored a new technique of generating human-like sentences using a neural network and then various linguistic features are extracted to predict the authorship. the proposed model with an accuracy of 97.2% can predict the true author successfully.

AA for a very lengthy corpus is a tedious job, using a reduction model [19], the size of the candidate authors can be reduced. Doc2Vec reduction was found to be efficient compared to other models used for reduction. It is observed that Reduction in the candidate authors set and the corpus didn't significantly affect the performance of the AA model. Reduction of authors set with a minimum of 10% and a maximum of 90%, the model achieved 99% and 50% accuracy respectively. Lexical feature extraction is easy compared to semantic or content based features, but semantic features are more realistic. Researchers in [20] developed a content based model for AA by considering authors from different domains and also datasets with different genres. The proposed model learns the sentences from POS tags and the system is trained and tested with RNN, the model was able to achieve maximum accuracy of 78% accuracy on the PAN dataset. AA on Persian historical and literary works is explored in [21]. The authors have used a modified four parts of deep convolutional neural networks architecture and attention mechanism. The model outperforms other approaches with 72.59% accuracy.

Authors [22] have tried to predict the owner of the e-mail which has some dispute contents, a user with fake mail ids can write unacceptable contents that may damage the reputation of a person/ company. Prediction is mainly based on reasonable hypotheses; authors have strived to develop a mathematical model to successfully address this problem by combining the Analytic Hierarchy Process with SVM. Experimental findings demonstrate that the accuracy is greater than 95%.

In the proposed work Syntactic features are extracted from Kannada articles to understand the author's writing style. The basic concepts of POS tagging and the different methodology to implement it is discussed in [23]. Authors have served the complete POS tagging information for beginners to carry out research in this domain. They concluded that deep learning algorithms are more powerful compared to traditional methods for the English language.

The authors used deep learning methodologies [24] like RNN and LSTM to assign POS tags to the annotated Kannada words and achieved 81% accuracy. The limitation of this work is in getting the clear dataset in the required format since the same words are spoken and written in different ways due to this one word can have different inflections. This leads to ambiguity in assigning the POS tags.

The authors have explored both the Hidden Markov chain method and conditional random fields algorithms [25] to assign POS tags to the Kannada words. They achieved 79% and 84% for both methods respectively. The model suffers due to cross domain dataset that is a dataset with different categories.

POS tags are assigned after analyzing each word in the text, authors have demonstrated POS tagging [26] using machine learning algorithms and deep learning algorithms. One of the machine learning algorithms SVM outperforms deep learning techniques with 85% accuracy. The lack of a clean dataset is the only demerit in this work.

Markov chain algorithm again proved to be efficient for a small Kannada dataset [27] of 18,000 words. researchers achieved 95% accuracy but their performance declines as the dataset increased.

Sindhi is one of the oldest languages and not much work have been done in the field of text processing [28], authors have designed rule based approach to assigning POS tags and they were able to assign POS tags successfully on 624 words. Performance comparison has not been focused on since there is no state of art models available for this language.

A large volume of data is flowing over the twitter media, and analyzing the data based on their POS tags are experimented with by the researchers [29]. They have used various classification algorithms to efficiently assign the POS tags to the Malay corpus. SVM yields a maximum accuracy of 95% compared to other algorithms. This approach can be implemented for various categories of Malay words.

Authorship attribution becomes a very important issue in today's time due to the increase in identity theft crimes. The text domain is in ranges from science, art, to philosophy-related texts. It is observed from the survey that, feature extraction plays a huge role in finding the source of a text (Lexical, Semantic and Syntactic features). Language Models (word/ character) and vocabulary of an author also are important parameters in performing the AA task. Few researchers have experimented on both Instance and Profile-based approaches for both global languages and a few Indian local Languages which include short and long texts. For the majority of the AA tasks, the deep learning technique [23] proved to be efficient but can't be claimed as a standardized technique since ML algorithms also outperformed well for other data samples.

There is a research gap in this domain for the Kannada language and this can be used as an opportunity by the interested researchers.

III. METHODOLOGY

The overall description of the proposed work is given below:

1) Let A be the author set $\{a_i, a_{i+1}, \ldots a_n\}$.

2) Let D be the document set $\{d_i, d_{i+1}, \ldots d_n\}$ written by the author ai, such that $d_i \in a_i$.

3) Let S be the sentences in a document d_i $S = \{s_i, s_{i+1}, \ldots s_n\}$ such that $s_i \in d_i$.

4) Let T be the set of POS tags $T=\{t_i, t_{i+1}, \ldots t_n\}$ for a sentence created using the CRF algorithm such that $T \in s_i$.

5) The model extracts semantic features using N-gram technique as $\{t_1, t_2, t_3\} \ldots . \{t_i, t_{i-1}, t_n\}$ where it is a POS tag $\in T$.

6) Let A_t be the anonymous text during the testing phase, the authorship model extracts the POS features of the A_t and compares them with the extracted features and predicts the probability of a true author.

A. Dataset Source and Collection

The primary source of the dataset is the internet, from that we selected articles from the Kannada blogs/ websites, e-articles, e-books from Kannada Sahitya Paritshath which is a government-based website and also from other popular Kannada websites. Few authors have contributed their articles based on the request. Totally 500 documents from 50 authors have been considered. It is a cross-domain dataset that each author has written articles on various categories which include: Life skills, philosophy, folk, children's stories, politics and sports. The proposed work is implemented based on the POS and sequence model. The AA is based on the POS tagging to extract the hidden semantic meaning of the text, since there is no open access POS tagging application available on the internet, we created a POS model [24], and the summary of the overall implementation is briefed below:

- Preprocess the dataset by tokenizing the documents (articles/stories/poems) of each author and creating an array of sentences.
- Pass the tokens to a POS tagger developed using the CRF model [25,26]. For the sake of better results, the number of parts of speech is reduced and more generalized, then run the bag of words code into the POS tagger and assign a tag to each word.
- Created embedding vectors, since the model understands only the numbers, not the text.
- Designed a sequential neural network with two LSTM layers, dense layers, activation and dropout layers.
- Train and test the model by running each document sentence-wise.

POS tags are assigned to all the tokens of the author's samples, this gives information about the semantic structure of the language that is the author's usual way of using words to form a sentence. This knowledge helps the model to understand the context of the Kannada words used in a sentence. A word can have different meanings so different POS tags [27,28], the tagger in this case will help a reader to understand the correct meaning of the words based on the tags. The proposed work uses a CRF classifier for assigning the POS tags to the Kannada tokens. The overall implementation of POS tagging in the proposed work is given in Fig. 2.

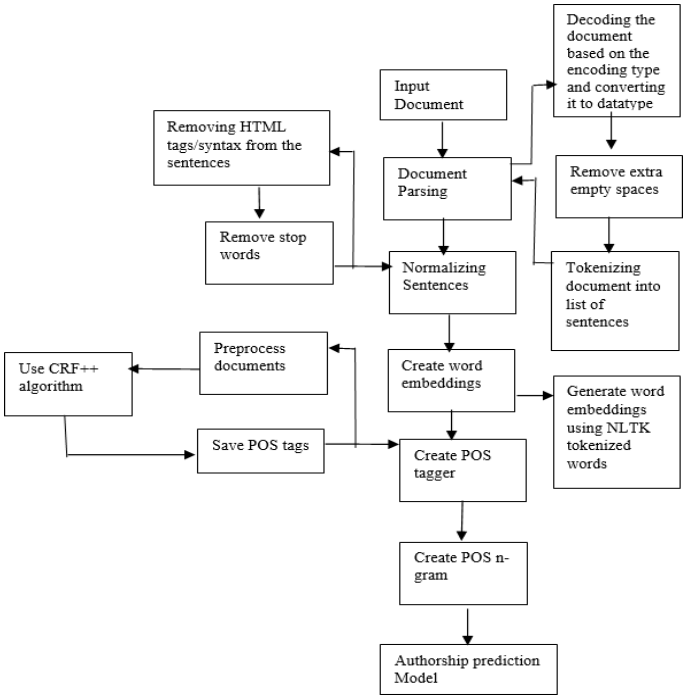

Fig. 2. Working Flow of POS Authorship Model.

B. Document Parsing

Document parsing, the first stage consists of three distinct steps.

- Parsing the text document: To build a POS tagging, a separate dataset prepared by the International Institute of Information Technology (IIIT), Hyderabad (IIIT Hyderabad LTRCMT-NLP Lab) is used for training and testing the POS model and Kannada scripts are encoded in the UTF-8 format. Each document will have to be read in the same format to preserve the text. PDF files were not used because no existing tool parsed PDF files with Kannada text in them. The text documents parsed act as the raw source on which various pre-processing steps are carried out:
- Data cleaning: The raw text extracted from the parsing stage is analyzed to check if there are any English words present in them. This stage is executed using the understanding of the range of Kannada words in the UTF-8 encoded format and regular expression. Attempts to remove the stop words were made but not considered for the final evaluation as they would prove to be important in POS detection.
- Tokenization: The text after the previous stage is tokenized into indexes so it can be used later. This process is carried out using out-of-the-shelf methods provided by the Keras API tokenization method. OOV token is used to make sure non-existent words in the dictionary can be marked during model evaluation. Sample output for the same is shown in Fig. 3.

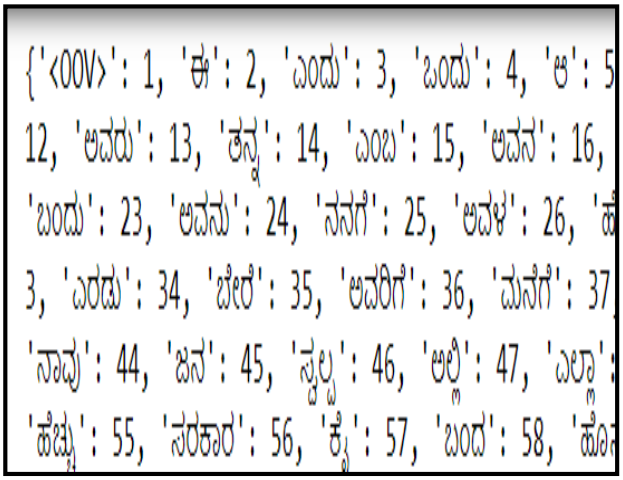

Fig. 3. Sample Tokenized Kannada Words from the Article.

C. Normalize Sentences

Once the data is parsed and cleaned for POS tagging. Normalizing the sentences on the other hand is used to preprocess the dataset used to train the POS model [11]. The dataset used for POS training comes in HTML format and thus, HTML tags (start tags, end tags, new line and parentheses) are removed to bring the data into raw text format. The raw text is then preprocessed to create a tag set where an index and a Part of Speech are attached to each word in a sentence. The first word of the sentence has index one and the index keeps increasing until a delimiter is found. The first word of the next sentence starts with index 1 again. This tag set contains complex parts of speech attached to each word. To reduce the number of classes for the classification model, we reduce the part of speech to its roots (11 categories: noun, verb, pronoun, intensifier, conjunction, adjective, demonstrative, quantifier, adverb, particle and punctuation). Table I shows the corresponding labels used in the datasets for various POS tags and Fig. 4 shows the output of the POS tagger.

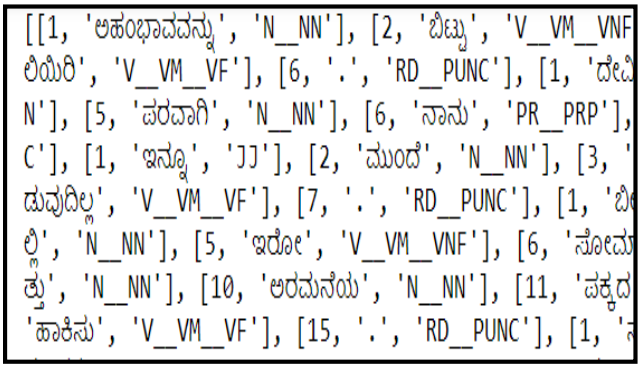

Fig. 4. POS Tagging for the Tokenized Words.

D. Creating the embeddings

The identified tokens are used to create a word embedding matrix using the off-the-shelf Language Tokenizer [12]. This produces a 1*400 vector for each word and an embedding matrix is created by stacking the vectors of each word based on their index from the tokenizer. Language_tokenizer( ) returns two vectors for certain words as it recognizes the root word and the suffix as two different words. The root word is assumed to have the maximum importance and is retained while the vector for the suffix is discarded. This gives a 27046*400 dimensions word embedding for the entire dictionary which is used as the first layer in the Sequence model.

TABLE I. KANNADA POS TAGS

Sl. No	POS tags in Kannada	Label
1	ನಾಮಪದ Noun	NP-Noun
2.	ಸರ್ವನಾಮ Pronoun	PR
3.	ಕ್ರಿಯಾಪದ MainVerb	V_ _VM_ _VF
4.	ತೀವ್ರಗೊಳಿಸುವಿಕೆ intensifier	RD_ _INTF
5.	ಸಂಯೋಗ conjunction	CC_ _ CCS
6	ವಿಶೇಷಣ adjective	JJ
7	ಪ್ರದರ್ಶಕ Demonstrative	DM DMD
8	ಪರಿಮಾಣಕಾರಕ Quantifier	QT_ _QTC
9	ಕ್ರಿಯಾವಿಶೇಷಣ adverb	RB
10	ಕಣ Particle	RP_RPD
11.	ವಿರಾಮಚಿಹ್ನೆ Punctuation	RD_PUNC

E. Create POS Tagging

The tag set generated in the normalize sentences section is used here. Each word in the POS tag set is processed to contain some features to use in the classification algorithm. The features considered are:

- The word itself.
- The length of the word.
- First 4 letters of the word.
- First 3 letters of the word.
- First 2 letters of the word.
- Last 4 letters of the word.
- Last 3 letters of the word.
- Last 2 letters of the word.
- Is the word a punctuation..
- Surrounding information
  - If the word is not the first word, the above mentioned features of the previous word.
  - If the word is not the second word, the features of the word that are two positions behind.
  - If the word is not the last word, above mentioned features of the next word.
  - If the word is not the last second word, the features of the word are two positions ahead.

The above features combined with N-gram predict the POS tag for a word based on the POS of the previous and the next word of that given word.

The performance of the POS tagger is given in Table II. With the proposed POS model, 91% accuracy is achieved.

TABLE II. PERFORMANCE OF THE PROPOSED POS MODEL

POS tags	Precision	Recall	F1 Score	Support
N	0.843	0.956	0.896	614
V	0.955	0.922	0.938	502
RB	0.444	0.316	0.369	38
CC	0.901	0.839	0.869	87
RD	1.000	0.997	0.998	317
JJ	0.806	0.532	0.641	47
DM	0.927	0.905	0.916	42
RP	0.647	0.333	0.440	33
PR	0.964	0.931	0.948	350
PSP	0.333	1.000	0.500	1
QT	0.913	0.808	0.857	26
Micro average	0.910	0.911	0.910	2057
Macro average	0.794	0.776	0.761	2057
Weighted average	0.909	0.911	0.907	2057
Final F1 score on the test dataset	0.9066			
Accuracy of the test dataset	0.9101			

The proposed POS model's performance is compared with those of other models identified during the survey. We were able to achieve a decent accuracy of 91% accuracy and a 90% F1 score since the model employs a clear dataset. The performance of the POS model is seen in Table III. The CRF model is one of the top models for assigning POS tags for Kannada words, according to the survey.

TABLE III. PERFORMANCE ANALYSIS OF VARIOUS POS MODEL

Reference No	Method employed	Accuracy in %
[24]	RNN+LSTM	81
[25]	Hidden Markov model CRF	79 84
[26]	SVM	85
[27]	Markov chain	95
[29]	SVM	95
Proposed model		**91**

F. Authorship model using Bi-LSTM

Once the author's dataset is prepared and the POS model assigns the tags for each token in the dataset, the next stage is, running the proposed authorship model. It is a classification problem and Bi-LSTM is employed to perform the AA task. Bi-LSTM model [13] reads the author's text in both directions and understands the syntactic features, especially how the sentence is formed using the POS tags. The Bi- LSTM architecture used for the proposed work is shown in Fig. 5 and the simplified architecture is shown in Fig. 6.

- Layer 1 composing 15 Bi-Directional LSTM Units: A Bidirectional LSTM, or Bi-LSTM, is a sequence processing model that consists of two LSTMs [10]: one taking the input in a forward direction, and the other in a backward direction. Bi-LSTMs effectively increase the amount of information available to the network, improving the context available to the algorithm (for example, knowing what words immediately follow and precede a word in a sentence).
- Layer 2 stacked on top of layer 1 consisting of 15 Bi-Directional LSTM Units.
- Batch Normalization for the sequence: It is a process to make neural networks faster and more stable by adding extra layers to a deep neural network. The new layer performs the standardizing and normalizing operations on the input of a layer coming from a previous layer.
- A densely connected neural network layer.
- ReLU activation: A linear function that will output the input directly if it is positive, otherwise, it will output zero.
- 64 Dense units with the 'ReLU' activation and a dropout of 0.5 32 Dense units with the 'ReLU' activation and a dropout of 0.5.
- Dropout: Dropout is a technique used to prevent a model from overfitting. Dropout works by randomly setting the outgoing edges of hidden units (neurons that make up hidden layers) to 0 at each update of the training phase.
- Sigmoid activation: A sigmoid function is a bounded, differentiable, real function that is defined for all real input values and has a non-negative derivative at each point and exactly one inflection point.

The model is compiled with the following hyperparameters:

- Loss: Binary cross entropy compares each predicted probability to actual class output, which can be 0 or 1. It then calculates the score that penalizes the probabilities based on the distance from the expected value. That means how close or far from the actual value.
- Optimizer: Stochastic Gradient Descent with a learning rate of 0.001 and momentum of 0.9. It attempts to find the global minimum by adjusting the configuration of the network after each training point. Instead of decreasing the error, or finding the gradient, for the entire data set, this method merely decreases the error by approximating the gradient for a randomly selected batch (which may be as small as a single training sample). In practice, random selection is achieved by randomly shuffling the dataset and working through batches in a stepwise fashion.
- Prediction: Each of the text documents is preprocessed similarly during the training and validation sets.

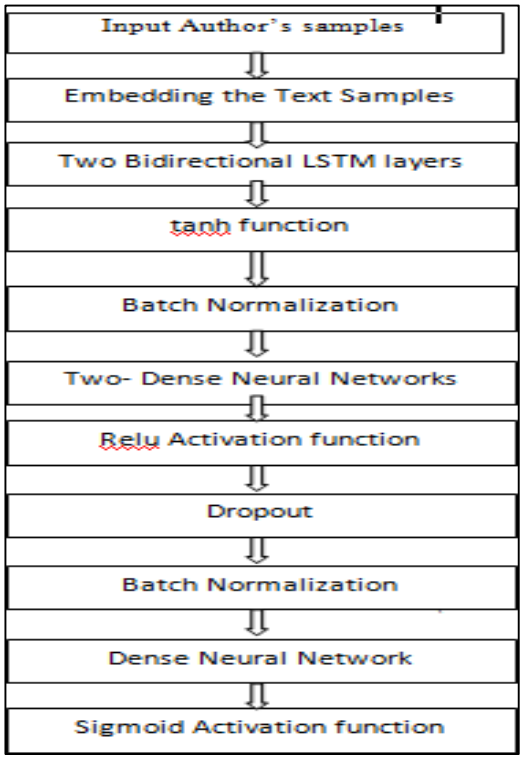

Fig. 5. Overall Bi- LSTM Architecture of the Proposed Model.

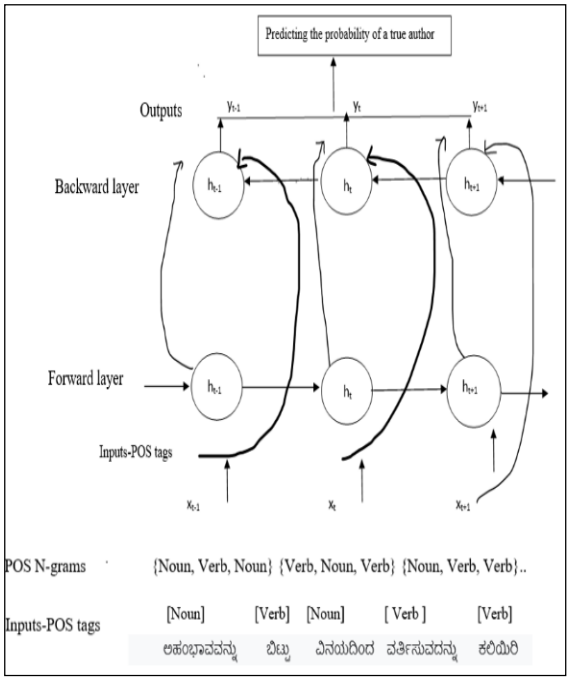

Fig. 6. The Architecture of the AA Model using Bi-LSTM.

All the documents with the labeled author's names are trained but testing is done in a different approach, that is 10 datasets are created such that, each dataset comprises articles written by five different authors so a total of 50 author's 10 documents are collected. Table III shows the dataset preparation for the proposed model. Finally, 500 articles are in the dataset, it is split into 70:20:10 for training, testing and validation respectively. The model is trained using the N-gram approach by extracting the POS tag features of the authors. During the testing phase, if author 1's document is passed, then Model 1 is supposed to respond positively to the statements of author 1 and should predict 0 for authors 2,3,4 and 5's documents. Similarly, model 10 contains articles written by the authors 46,47,48,49 and 50. After predicting the author for each sentence in each document, the ratio of positive to negative statements is calculated. We used the 1v5 method for testing rather than having 5 neurons that associate a document to an author with a certain probability since this model produced more promising results.

IV. Results and Discussion

The performance of 5 author sets of 10 models is tabulated based on the N-grams of POS tags. One of the metrics to measure the performance of the proposed model is the count of positive and negative statements.

The overall loss rate of all the dataset models is shown in Fig. 7, the loss rate is reduced after several epochs which indicates that the model is learning the writing style of the authors in a better way. After testing the author's samples with the different combinations, we obtained the accuracy of all ten author sets and tabulated them in Table IV and Fig. 8 to indicate the same.

We also observed that for a few articles authors are mispredicted. The performance is mainly depending on the efficiency of the POS tagger [12]. POS tagger is working fine with the training set but has average performance for the authorship dataset. The proposed work uses cross domain authorship that is an author can write political as well as scientific articles since the writing style differs, it has an impact on the accuracy and the N-gram technique extracting the POS tags is not sufficient to extract the writing style, N-gram combined with other stylometry features may work better. The performance of the model deteriorates when the anonymous text document is tested against the articles of all 50 authors. The accuracy was 10%-12% so a 1v5 approach is used. In the 1v5 approach, each time the dataset model contains different authors' articles.

TABLE IV. Author Wise Accuracy

Dataset Models	Accuracy
Model-1	77%
Model-2.	78%
Model-3	77%
Model-4	78%
Model-5	81%
Model-6	77%
Model-7	81%
Model-8	78%
Model-9	72%
Model-10	79%
Average Accuracy	77.8%

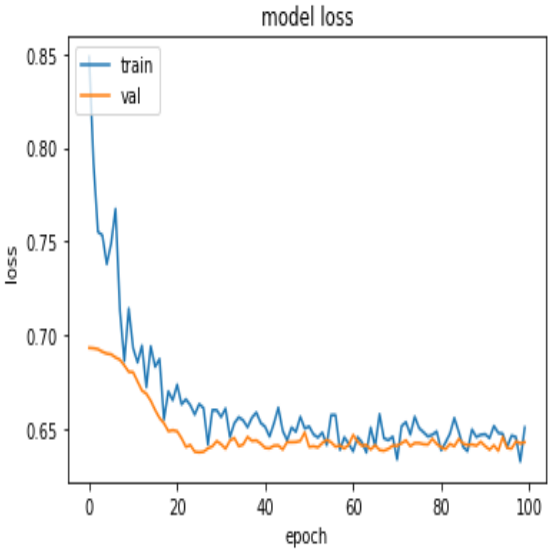

Fig. 7. Loss Rate.

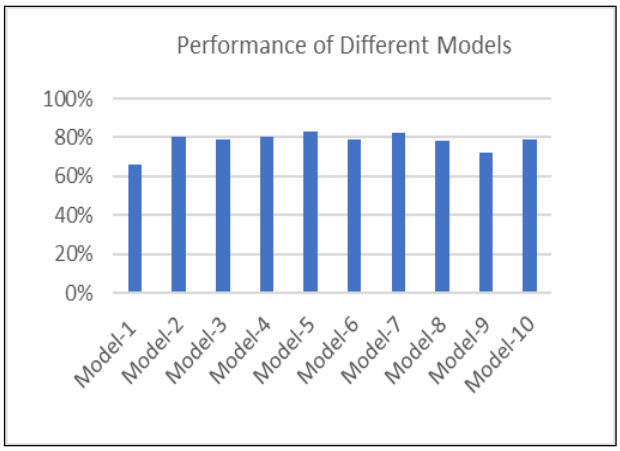

Fig. 8. Accuracy of All the Dataset Models.

Using LSTM one more approach called the sequence model is developed. This experimental model was used without convincing results. The text source created at the end of step 3 as shown in Fig. 5 is used for this model. After splitting the dataset as train, test and validation sets like the BI-LSTM model. All 50 authors were used for this model and the multi-class classification strategy was employed here. The prediction vector was in the form of a one-hot encoded vector where the index of the corresponding author was marked as 1 and the rest as 0. The sets from the previous stage are converted to indices using the tokenization from stage 4 of the Data Processing module. Both sentence level and document level sequences were considered with lengths 15 and 2500 respectively but this model yields overall accuracy of 15%.

A. Performance Comparison of Different Techniques

To assess the performance of our model, we have compared it with other AA models developed for other languages since no models are available for the Kannada languages. Table V shows the accuracy obtained by other AA models. We observed that deep learning based models performed well with more than 90% accuracy, but we obtained 77.8% accuracy for the proposed work. Also, classification algorithms perform better in some cases. The reasons for the moderate performance of the proposed model are identified and listed below:

- The POS model is tested and trained successfully on a labeled dataset, but when it is used on a real authors dataset, it performs moderately since certain Kannada words have several meanings, which causes ambiguity.
- In the proposed work a cross-domain dataset is considered which has a variety of categories like life skills, science and health, sports, etc. It is common in foreign languages but is new to the Kannada language.
- A good dataset of more than 10,000 articles improves the model's performance, however building such a huge Kannada dataset is a challenging one.

TABLE V. PERFORMANCE COMPARISON OF AA MODELS

Reference No	Technique used	Accuracy / F1 Score in %
[9]	Artificial Neural Network	75.46
[10]	N-gram	61.1
[11]	Life Like Network Automata using Life-Like rules	70-75
[12]	SVM algorithm with linear kernel	68
[14]	SVM	96
	Deep learning	93
[16]	Convolutional Neural Network	98
[17]	BERT model	90
[18]	Neural Networks	97.2
[19]	Doc2vec Reduction model	99
[20]	Recurring Neural Networks	78
[21]	Convolutional Neural Network	72.59
[22]	SVM	95
Proposed Model	**Bi-LSTM**	**77.8**

B. Directions for Future Work

More powerful style markers can be introduced which can be used to classify a wide range of authors [30]. Right now, this work can classify 50 authors. The next step can be the development of general rules that should apply to almost every author. A near perfect classification can be achieved by training a meta-learner that will use both neural networks and decision trees, this is required because neural networks consider different sets of features than decision trees.

A combination of different sets of features may be tried to see if there exists a set that can be used by all learning algorithms. Also, feature extraction can be improved by using more powerful natural language tools. Features selection is an important criterion and the graph-based neural networks [14] can be used in the future where the network will itself select the features which contribute the most to the output.

V. CONCLUSION

The model currently stands with an accuracy of 77.8%. According to our knowledge, the proposed model is a novel technique and a challenging one in recognizing the writing style of a Kannada author. The proposed work aims to increase the accuracy by tweaking the model and if possible, implementing the same model by extracting features other than syntactic. It is understood that lexical features, word length and sentence length do not often have enough descriptive power for any model to assign a document to an author with a sense of certainty. We aim to pursue further research in detail to pick only those features that will yield good results. We believe semantics is one of those.

REFERENCES

[1] Shreyashree, S., Sunagar, P., Rajarajeswari, S. and Kanavalli, A, "A Literature Review on Bidirectional Encoder Representations from Transformers," Lecture Notes in Networks and Systems, Springer, Singapore, vol 336,pp. 305-320, Jan 2022.

[2] Pramod Sunagar and Anita Kanavalli, "A Hybrid RNN based Deep Learning Approach for Text Classification," International Journal of Advanced Computer Science and Applications(IJACSA), Vol13(6), pp. 289-295, July 2022.

[3] B.V.Dhandra and M.B.Vijayalaxmi, "A Novel Approach to Text Dependent Writer Identification Of Kannada Handwriting", Procedia, CS, Volume 49, pp. 33-41, 2015.

[4] Praveen Bangarimath and DeepaBendigeri, "Writer Identification using Texture Features in Kannada Handwritten Documents", IJCA Proceedings on National Conference on Electronics, Signals and Communication, Vol 6 13, pp.5-8, 2018.

[5] Fakhraddin Alwajih, Eman Badrand and Sherif Abdou, "Transformer-based Models for Arabic Online Handwriting Recognition," International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 13, No. 5, pp. 898-905, 2022.

[6] Chandrika C.P, Kallimani, J.S, "Authorship Attribution for Kannada Text Using Profile Based," Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications, Lecture Notes in Networks and Systems, Springer, Singapore, vol 237, pp. 679-688, Jan 2022.

[7] Chandrika C.P and Kallimani, J.S, "Instance Based Authorship Attribution for Kannada Text Using Amalgamation of Character and Word N-grams Technique", Distributed Computing and Optimization Techniques, Lecture Notes in Electrical Engineering, Springer Singapore, vol 903, pp 547–557, Aug 2022.

[8] Efstathios Stamatatos. "A Survey of Modern Authorship Attribution Methods," Journal of the American Society for Information Science and Technology, vol 60, pp 538–556, March 2009.

[9] Mohammad Al-Sarem, Abdullah Alsaeedi, and Faisal Saeed, "A Deep Learning-based Artificial Neural Network Method for Instance-based Arabic Language Authorship Attribution", International Journal of Advances in Soft computing and its Applications, ISSN 2074-852, Vol. 12, pp. 1-14, Dec 2020.

[10] Yaakov HaCohen-Kerner, Daniel Miller, Yair Yigal, and Elyashiv Shayovitz, "Cross-domain Authorship Attribution: Author Identification using Char Sequences, Word Uni-grams, and POS-tags Features," Notebook for PAN competition at CLEF 2018.

[11] Machicao J, Corrêa EA Jr, Miranda GHB, Amancio DR and Bruno OM, "Authorship attribution based on Life-Like Network Automata", PLoS One, Vol:13(3), DOI: 10.1371/journal.pone.0193703, March 2018.

[12] Andrea Bacciu, Massimo La Morgia, Alessandro Mei, Eugenio Nerio Nemmi, Valerio Neri, and Julinda Stefa, "Cross-Domain Authorship Attribution Combining Instance-Based and Profile-Based Features," A notebook for PAN competition at CLEF 2019.

[13] C. NamrataMahender, Ramesh Ram Naik and Maheshkumar Bhujangrao Landge, "Author Identification for Marathi Language," Advances in Science, Technology and Engineering Systems Journal, India, ISSN: 2415-6698, Vol. 5, No. 2, pp. 432-440.

[14] Aleksandr Romanov, Anna Kurtukova , Alexander Shelupanov, Anastasia Fedotova and Valery Goncharov, "Authorship Identification of a Russian-Language Text Using Support Vector Machine and Deep Neural Networks", Future Internet 2021, Vol: 13, pp. 1-16, Dec 2020.

[15] Aisha Khatun, Anisur Rahman, Md. Saiful Islam and Marium-E-Jannat, "Authorship Attribution in Bangla literature using Character-level CNN," 22nd International Conference on Computer and Information Technology (ICCIT), pp 1-5, 2019.

[16] Georgios Barlas and Efstathios Stamatatos, "Cross-Domain Authorship Attribution Using Pre-trained Language Models," Artificial Intelligence Applications and Innovations, IFIP Advances in Information and Communication Technology, Vol 583, DOI:https://doi.org/10.1007/978-3-030-49161-1_22, pp 255-266, May 2020.

[17] S. H. H. Ding, B. C. M. Fung, F. Iqbal and W. K. Cheung, "Learning Stylometric Representations for Authorship Analysis," IEEE Transactions on Cybernetics, DOI: 10.1109/TCYB.2017.2766189, vol. 49, no. 1, pp. 107-121, 2019.

[18] Michael Tschuggnall, Benjamin Murauer and Gunther Specht, "Reduce & Attribute Two-Step Authorship Attribution for Large-Scale Problems," Proceedings of the 23rd Conference on Computational Natural Language Learning, Hong Kong, China, pp. 951–960, Nov 2019.

[19] Fereshteh Jafariakinabad, Sansiri Tarnpradab and Kien A. Hua, "Syntactic Neural Model for Authorship Attribution," The Thirty-Third International FLAIRS Conference (FLAIRS-33), pp.1-6, 2020, May 2020.

[20] Ehsan Reisi1 and Hassan Mahboob Farimani, "Authorship Attribution in Historical and Literary Texts by A Deep Learning Classifier," Journal Of Applied Intelligent Systems &Information Sciences, Vol 1. Issue 2, pp. 118-127, December 2020.

[21] Suhad A. Yousif, Zainab N. Sultani, Venus W. Samawi, "Utilizing Arabic WordNet Relations in ArabicText Classification: New Feature Selection Methods," IAENG International Journal of Computer Science, Vol 46:4, 2019.

[22] Qinghe Zheng, Xinyu Tian, Mingqiang Yang and Huake Su, "The Email Author Identification System Based on Support Vector Machine (SVM) and Analytic Hierarchy Process (AHP)", IAENG International Journal of Computer Science, vol 46:2, pp.1-14, May 2019.

[23] Chiche, A and Yitagesu, B, "Part of speech tagging: a systematic review of deep learning and machine learning approaches," J Big Data, Vol9(10), United Kingdom, pp.2-25, Jan 2022.

[24] Rajani Shree, M., Shambhavi, B.R , " POS Tagger Model for South Indian Language Using a Deep Learning Approach", Lecture Notes in Electrical Engineering, Springer, Singapore, vol 828, pp.26–30, Jan 2022.

[25] Shambhavi, Ravi and P Ramakanth, "Kannada Part-Of-Speech Tagging with Probabilistic Classifiers," International Journal of Computer Applications, Vol 48. pp.26-30, June 2012.

[26] Shriya Atmakuri, Bhavya Shahi, Ashwath Rao B and Muralikrishna SN, "A comparison of features for POS tagging in Kannada," International Journal of Engineering & Technology, vol 7, pp.2418-2421, 2018.

[27] Saritha Shetty and Savitha Shetty, "Text pre-processing and parts of speech tagging for Kannada language," Journal of Xi'an University of Architecture & Technology, vol 11, pp 1286- 1291,2020.

[28] Irum Naz Sodhar, Abdul Hafeez Buller, Suriani Sulaiman and Anam Naz Sodhar, "Word by Word Labelling of Romanized Sindhi Text by using Online Python Tool" International Journal of Advanced Computer Science and Applications(IJACSA), vol 13(8), pp.262-267, 2022.

[29] Siti Noor Allia, Noor Ariffin and Sabrina Tiun, "Improved POS Tagging Model for Malay Twitter Data based on Machine Learning Algorithm", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 13, No. 7, pp 229-234, 2022.

[30] Patrick Juola, "Future Trends in Authorship Attribution," IFIP International Federation for Information Processing, Advances in Digital Forensics III, Volume 242, pp. 119-132.