PREDICTING GENDER OF BRAZILIAN NAMES USING DEEP LEARNING

A PREPRINT

Rosana C. B. Rego
Program in Electrical and Computer Engineering
Federal University of Rio Grande do Norte, Brazil
rosana.rego@ufrn.edu.br

Verônica M. L. Silva
Department of Engineering and Technology
Federal Rural University of Semi-Arid, Brazil
veronica.lima@ufersa.edu.br

June 21, 2021

ABSTRACT

Predicting gender by the name is not a simple task. In many applications, especially in the natural language processing (NLP) field, this task may be necessary, mainly when considering foreign names. Some machine learning algorithms can satisfactorily perform the prediction. In this paper, we examined and implemented feedforward and recurrent deep neural network models, such as MLP, RNN, GRU, CNN, and BiLSTM, to classify gender through the first name. A dataset of Brazilian names is used to train and evaluate the models. We analyzed the accuracy, recall, precision, and confusion matrix to measure the models' performances. The results indicate that the gender prediction can be performed from the feature extraction strategy looking at the names as a set of strings. Some models accurately predict the gender in more than 90% of the cases. The recurrent models overcome the feedforward models in this binary classification problem.

1 Introduction

Prediction problem refers to a wide class of problems in which the goal is to perform an inference based on situational plays and statistical-based models. However, in some situation is not simply make an inference. In this way, artificial intelligence (AI) algorithms can be applied to produce a prediction in which the problem goal is to provide the correct label (e.g., prediction or output) to an instance (e.g., features or input). One of the well-known prediction problems is gender inference.

In some applications, infer the gender of a person is necessary, such as investigations of psychological, anthropological, and sociological research questions [1]. But how could we infer gender? Indeed there are many research efforts all over the world in this field that are centered in many directions, such as NLP [2, 3], computer vision [4, 5], and image processing [6]. Yet, not all data is complete with the gender information.

The completeness of the database used in training and inference models and algorithms is essential for applications of AI. Sometimes these data do not have important information for the problem, such as lack of gender information. In order to solve this problem, it is possible to use pre-existing information to predict other information needed to fill the database. Using some basic logic and machine learning, we can infer gender by taking the name and compares it against a list of names associated with either a male or female gender, as explained in the following papers [7, 8, 9].

Gender prediction from other attributes is a technique already used in artificial intelligence applications. Many works predict gender from images of people's faces, such as [10, 11, 12, 13, 14, 15]. In the paper [10], the authors studied the deep residual networks of residual networks (RoR) capability to achieve gender prediction by analyzing images. In the same way, Wang et al. [11] explored the deep convolutional neural networks (CNN) capability to perform gender recognition by examining images. Venugopal et al. [12] applied the support vector machine (SVM) algorithm to classify children's gender from images. Also, using SVM, Kuehlkamp and Bowyer [14] predicted gender by processing images. Furthermore, the authors compared SVM with the CNN network. In the works [15], the authors explored the image processing techniques through SVM classifier.

Other researchers proposed gender prediction using natural language processing [16, 17, 18]. For instance, Vashisth and Meehan [16] explored some NLP techniques, such as bag of words, word embedding, logistic regression, SVM, and Naïve Bayes, to infer the gender of a person based on Twitter data. In the work [17], the authors proposed two Long Short Term Memory (LSTM) to predict the effect of Gana on personal names and perform gender inference. Mamgai et al. [18] applied logistic regression, random forest, bag-of-words, and LSTM-CNN to infer gender and language variety of authors. They concluded that the LSTM-CNN model works better for language variety tasks and bag-of-words gives good accuracy for gender prediction. Motivated by the above discussion, we proposed some character-level deep learning models based on NLP to infer a person's gender by the first name.

Deep learning is a subset of machine learning. It has been widely applied to deal with prediction problems [19]. Deep learning, also known as deep neural network, can be classified into deep feedforward neural networks (DFNN) or just multilayer perceptron (MLP), CNN, and recurrent and recursive neural networks (RNN) [20]. Based on the capability of the deep models, we implemented five character-level deep neural network models as MLP, CNN, RNN, bidirectional long short term memory (BiLSTM), and gated recurrent unit (GRU) to realize the gender prediction through the first name. Thus, the main contributions of this work are pre-processing names using character encoding via one-hot encoding technique and predicted gender with deep neural network models based on the person's first name. Moreover, we examined two categories of deep learning models, such as feedforward (MLP, CNN) and recurrent (RNN, GRU, BiLSTM) networks for gender classification problems. The model performance is examined through accuracy, recall, precision, and confusion matrix.

2 Related Work

There has been some research that uses the name of a person to infer gender [7, 8, 9, 21, 22, 23]. Panchenko and Teterin [7] first proposed gender prediction using the full name of a person. They consider Russian full names and used a 2-regularized Logistic Regression model for inference, which given accuracy up to 96%. In the work [21] a statistical machine learning to named entity recognition and classification system for the Kannada Language is proposed. The system includes identification of proper names in texts and classification of those names into a set of categories such as person names and organization names. In addition, a Naive Bayes model is evaluated for Kannada Indian names resulting in a 10-fold cross-validation accuracy of 77.2%.

A logistic regression model is applied on Indonesian names for gender inference in [22] resulting in an accuracy of 83.7%. Logistic regression is further combined with a CNN modeled to infer gender through profile photos resulting in an improvement in accuracy to 98.6%. Also, in [8], a logistic regression model is applied to predict the gender of written and spoken Chinese names. The authors verified that having both modalities available (i. e., written and spoken) did not significantly increase the identification of a name as female or male compared to the written-only condition.

Machine learning models such as SVM, Naive Bayes, Random Forest are used in [23] to identify gender through the first name and nickname of Thai Facebook profiles. The results obtained using word tokenization present accuracy of 65% for the base models and more than 90% for the combined models. In [9], character-based machine learning models are compared with models based on content information for gender prediction. Character-based models performed better for the trained data. In this work, also some deep learning architectures are tested. They resulted in performances as good as the base models combined with feature engineering. Moreover, differently from the previous works, we compared the five deep learning models. These models predicted the gender well, considering only the first name.

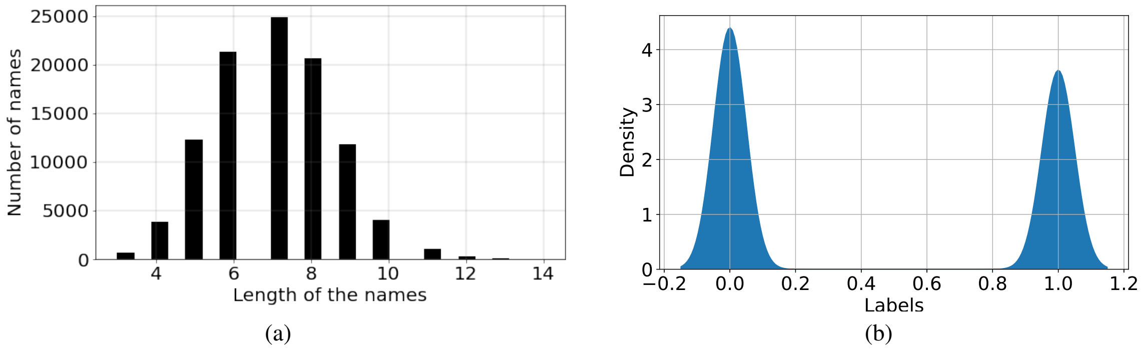
3 Dataset and Pre-processing

The dataset, the pre-processing techniques used, and the performance measures parameters are described in this section.

3.1 Dataset

The dataset consists of 100787 Brazilian names, which 54.82% are females names and 45.18% are males, based on 2010 CENSO data from Brasil.io[1] project. The distribution of the length of names based on the number of letters is depicted in Figure 1 (a). The label's distribution considering the gender (0 for female and 1 for male) is depicted in Figure 1 (b). We are particularly examining the first name of a person. Consequently, when we talk about a name, we are referring to the first name. Into the dataset, we have gender information, name, total frequency, group frequency, group name, and ratio. The first ten rows of the dataset are depicted in Table 1.

[1] https://brasil.io/dataset/genero-nomes/grupos/



(a) (b)

Figure 1: (a) Length of the names; (b) Labels distribution.

Table 1: Data frame first 10 rows.

Gender	Name	Total frequency	Group frequency	Group name	Ratio
M	AARAO	281	3526	ARAO	1.0
M	AARON	676	3442	ARON	1.0
F	ABA	82	5583	ADA	1.0
M	ABADE	57	57	ABADE	1.0
M	ABADI	73	116	ABADI	1.0
F	ABADIA	7565	7565	ABADIA	0.9832
M	ABADIAS	201	201	ABADIAS	0.7761
M	ABADIO	1550	1550	ABADIO	1.0
M	ABAETE	39	233	ADETE	1.0
M	ABD	23	23	ABD	1.0

A neural network can only discover patterns in numerical data, so it is needed to transform our data into numeric values with the process called encoding word.

3.2 Encoding words

The encoding word process consists in converts the letters into numbers. Then, the first step to encode the letter is to define a glossary that corresponds to all the single letters encountered in the alphabet and maps each letter of glossary to a number. We used the *word2vec* technique to encode the names, which used one-hot encoding [24]. Basically, we set a vector with zeros and a 1 which represents the corresponding letter included in the name and existing vocabulary/glossary. Our vocabulary has 28 letters and we set a maximum number of characters by name as 20.

Therefore, if we have the name "*ana*" as input, after the one-hot encoding, the name will be represented as

$$\begin{aligned}
a:&\quad [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0]\\
n:&\quad [0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]\\
a:&\quad [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0].
\end{aligned}$$

With this encoding process, our network models will be able to predict accurately the gender of names, as we showed in the numerical simulation section.

3.3 Performance measures

Because our problem is a binary classification (1 or 0, M or F), we selected the binary crossentropy as loss function to train each model. Therefore, the loss function is given by

$$L(p,q) = -\frac{1}{N}\left[\sum_{i=1}^{N} y_i \cdot log(p(y_i)) + (1-y_i)\cdot log(q(y_i))\right], \tag{1}$$

where y is the label (0 for female and 1 for male), $p(y)$ is the predicted probability of the gender being male for all N points, and $q(y_i) = 1 - p(y_i)$ is the predicted probability of the gender being female.

We adopted some performance measures to analyze the deep network models. The first one is the accuracy defined by the equation

$$accuracy := \frac{TN + TP}{TN + FP + FN + TP}, \tag{2}$$

where TP is true positive value, TN is true negative, FP is false positive, and FN is false negative value. Because accuracy is not always a sufficient performance measure, we calculated the recall, also known as the true positive rate given by

$$recall := \frac{TP}{FN + TP}. \tag{3}$$

Moreover, we also determined the precision defined by

$$precision := \frac{TP}{FP + TP}. \tag{4}$$

Through the examination of precision, recall, and accuracy, we can fully evaluate the effectiveness of the proposed deep learning models.

4 Deep learning models

We designed five deep neural network models: CNN, MLP, simple RNN, BiLSTM, and GRU.

4.1 Convolutional Neural Networks

Character-level CNN (char-CNN) can be used in natural process language as presented by [25]. We proposed a character-level ConvNet contains two convolutional layers with relu as activation function, two fully-connected layers, and we set sigmoid as output activation function. The suggested char-CNN architecture to gender names classification is displayed in Figure 2.

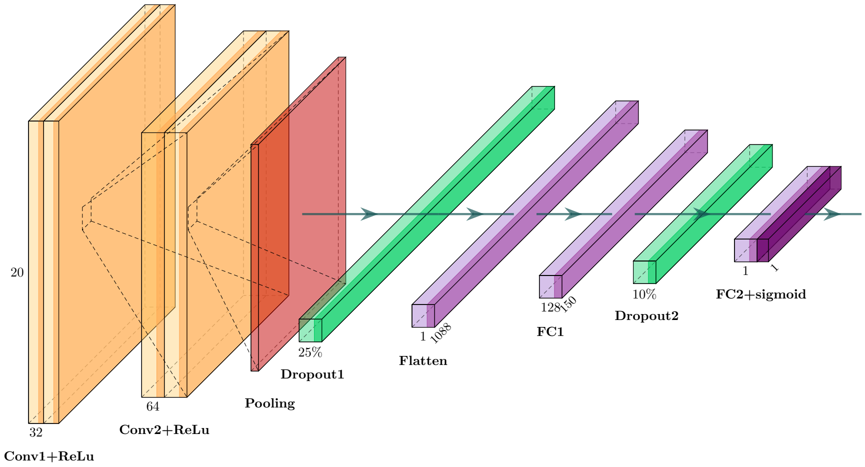


Figure 2: Convolutional neural network model.

4.2 Multilayer Perceptrons

A multilayer perceptron or deep feedforward networks learn deterministic mappings from input to output that lack feedback connections [20]. An MLP can classify data that is not linearly separable. In NLP, it can be applied to speech recognition and machine translation [26]. We implemented an MLP with two hidden layers. For the first hidden layer with 64 neurons, the relu activation function was considered. For the second hidden layer with 128 neurons, the softplus activation function was considered. For the output layer, we set the sigmoid activation function.

4.3 Recurrent Neural Networks

Basically, there are two functional uses of feedback or recurrent neural networks: associative memories and input-output mapping networks [27]. There are many architectures in the literature for recurrent networks, such as those based on a recurring input-output model, state-space, recurrent multi-layer perceptron, fully connected recurrent neural network (FCRNN), among others. All these architectures exploit the multi-layer perceptron nonlinear mapping capabilities [27]. But, in RNN the output depends not only on the current network input but also on the current or previous network outputs or states. They have important features not found in feedforward networks, such as the ability to store information for later use [20]. For this reason, recurrent networks are more powerful than non-recurring networks.

We implemented a simple RNN with 32 units, and the sigmoid activation function in the output. As shown in Figure 3, the RNN receives as input the sequence and classifies the name by gender.

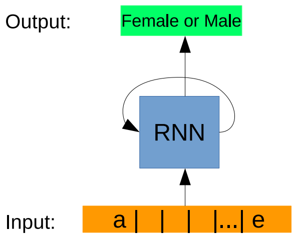


Figure 3: Simple RNN model.

4.4 Bidirectional Long Short Term Memory Networks

A Long Short-Term Memory is a popular RNN proposed by [28] to solve the problem of vanishing gradients that happens when we are training a simple RNN. A Bidirectional Long Short Term Memory Network is an LSTM modification in which the BiLSTM network scans at a particular sequence both from front to back and from back to front as depicted in Figure 4.

We designed a BiLSTM model with 64 units in each forward and backward layer. We added a dropout rate of 20% and $L2$ regularization penalty with a factor of 0.002, and we applied the sigmoid activation function in the output.

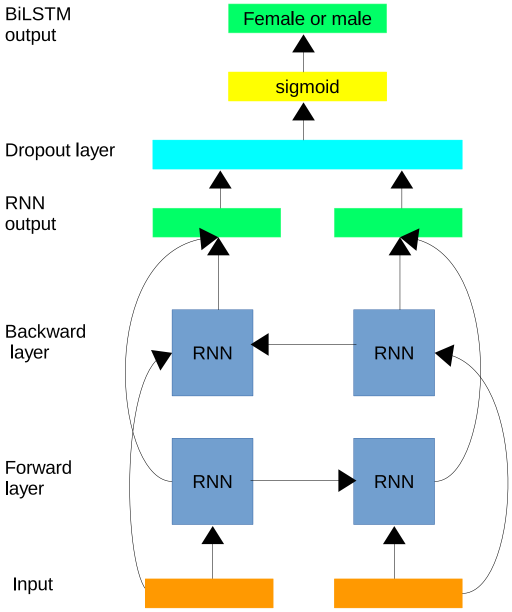


Figure 4: Bidirectional long short term memory model.

4.5 Gated recurrent units

The Gated recurrent unit neural network proposed by [29] has been created to solve the vanishing gradient problem, as LSTM. Both are designed similarly. To understand better the difference between GRU and LSTM see the paper [30].

We also built a GRU model with 32 units with the sigmoid activation function in the output to analyze with the BiLSTM model, since both are related.

5 Numerical simulation

All models were implemented using Keras 2.5 framework, Tensorflow 2, and python 3.6.9. We split our dataset into train data, validation data, and test data with 60%, 20%, 20% of the dataset, respectively. We used a learning rate (LR) schedule that uses an exponential decay schedule in the training process. We set initially $LR = 0.01$ and a batch size of 128. Our problem consists of binary classification, so for all models, the binary crossentropy loss was considered. Moreover, all the models were trained with Adam algorithm.

The accuracy achieved, with the test data, for all models is depicted in Table 2. Besides, the accuracy progression through epochs during the train and validation are shown in Figure 5. Based on the accuracy, we can infer that the recurrent neural networks perform better to predict the gender give a name. The models achieve accuracy greater than 90%, excluding the MLP model, which accomplishes 86.92%.

Table 2: Model-performance measures

Model	CNN	BiLSTM	RNN	MLP	GRU
Accuracy	92.67%	95.89%	93.85%	86.92%	94.80%
Recall	0.9211	0.9532	0.9378	0.8154	0.9332
Precision	0.9169	0.9556	0.9268	0.8860	0.9507

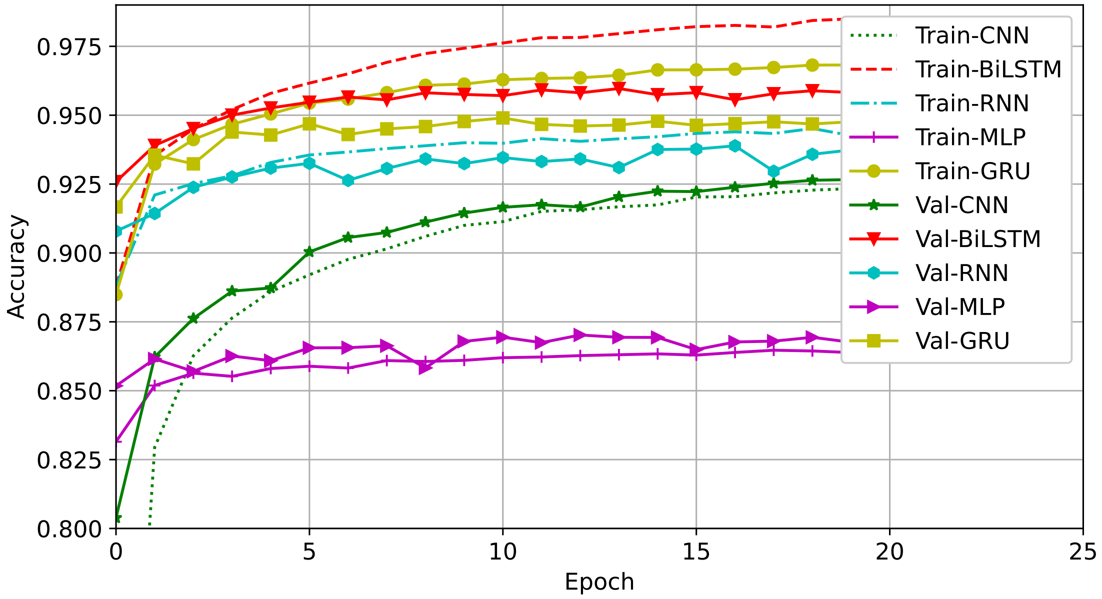


Figure 5: Accuracy versus epochs.

The values assumed by the loss function, during the train and validation, through epochs are displayed in Figure 6. In the MLP model, the values during the validation are less than the train this is because of the dropouts in our model, it is applied during training, but not during the validation. The same happens with the CNN model, but the validation loss is close to the train loss. For the BiLSTM and GRU model, we set an early stop at epoch 19 to avoid the lack of model generalization.

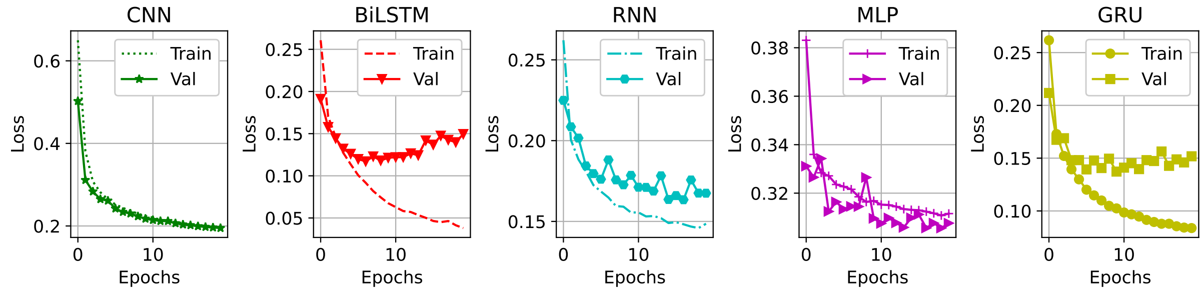


Figure 6: Loss versus epochs.

For evaluating the approaches, we used the confusion matrix to measure the performance of deep learning models, as shown in Figure 7. Interpreting the confusion matrix, we have the predicted labels on the x-axis and the real labels on the y-axis. The black and gray cells carry the number of names that the models accurately predicted, and the white cells contain the number of names that the models incorrectly predicted. Analyzing the confusion matrix, we can conclude that the recurrent models RNN, BiLSTM, and GRU performed more favorable outcomes. In other words, fewer confusions occur in the classification with these models. Indeed, this is what accuracy implies. But since accuracy is not regularly an adequate performance measure, we calculated the precision and recall, which is the true positive rate. The precision and recall are depicted in Table 2. By adopting the encoding word process previously described, all the models achieve considerable results, as shown by recall values. However, BiLSTM model with a precision of 0.9556 outperforms all other implemented models.

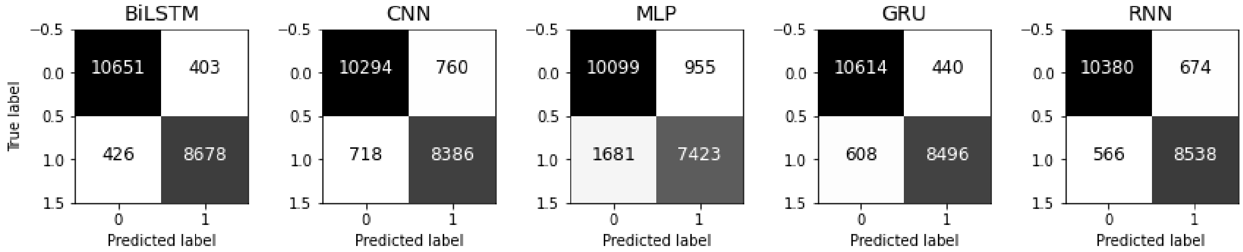


Figure 7: Confusion matrices.

6 Conclusion

We presented how deep neural network models can process and classify Brazilian names using encoding words, more specifically encoding characters. The deep network models presented are capable of successfully capture dependencies in the word vector. Through the validation, we concluded that the MLP model is suitable for classification prediction. However, during the comparison, the other models may appear more suited to the gender prediction problem. The benefit of using CNNs is their capacity to develop an internal representation. Nevertheless, because of the ability to deal with sequence prediction, recurrent models, such as RNN, BiLSTM, and GRU, achieve better results.

As future work, we will extend the dataset to people's names all over the world and analyzed how the recurrent deep models performers. Furthermore, we intend to compare the preprocessing required in the deep models, and their performs with some classical artificial intelligence algorithms, such as SVM, K - Nearest Neighbors (KNN), and Naive Bayes.

References

[1] F. Karimi, C. Wagner, F. Lemmerich, M. Jadidi, and M. Strohmaier, "Inferring gender from names on the web: A comparative evaluation of gender detection methods," in *Proceedings of the 25th International conference companion on World Wide Web*, pp. 53–54, 2016.

[2] M. Vicente, F. Batista, and J. P. Carvalho, "Gender detection of twitter users based on multiple information sources," in *Interactions Between Computational Intelligence and Mathematics Part 2*, pp. 39–54, Springer, 2019.

[3] A. I. Al-Ghadir and A. M. Azmi, "A study of arabic social media users—posting behavior and author's gender prediction," *Cognitive Computation*, vol. 11, no. 1, pp. 71–86, 2019.

[4] H. Ugail, "Secrets of a smile? your gender and perhaps your biometric identity," *Biometric Technology Today*, vol. 2018, no. 6, pp. 5–7, 2018.

[5] D. K. K. Galla, B. R. Mukamalla, and R. P. R. Chegireddy, "Support vector machine based feature extraction for gender recognition from objects using lasso classifier," *Journal of Big Data*, vol. 7, no. 1, pp. 1–16, 2020.

[6] M. Afifi and A. Abdelhamed, "Afif4: deep gender classification based on adaboost-based fusion of isolated facial features and foggy faces," *Journal of Visual Communication and Image Representation*, vol. 62, pp. 77–86, 2019.

[7] A. Panchenko and A. Teterin, "Detecting gender by full name: Experiments with the russian language," in *International Conference on Analysis of Images, Social Networks and Texts*, pp. 169–182, Springer, 2014.

[8] J. van de Weijer, G. Ren, J. van de Weijer, W. Wei, and Y. Wang, "Gender identification in chinese names," *Lingua*, vol. 234, p. 102759, 2020.

[9] Y. Hu, C. Hu, T. Tran, T. Kasturi, E. Joseph, and M. Gillingham, "What's in a name?–gender classification of names with character based machine learning models," *Data Mining and Knowledge Discovery*, pp. 1–27, 2021.

[10] K. Zhang, C. Gao, L. Guo, M. Sun, X. Yuan, T. X. Han, Z. Zhao, and B. Li, "Age group and gender estimation in the wild with deep ror architecture," *IEEE Access*, vol. 5, pp. 22492–22503, 2017.

[11] C. H. Nga, K.-T. Nguyen, N. C. Tran, and J.-C. Wang, "Transfer learning for gender and age prediction," in *2020 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-Taiwan)*, pp. 1–2, 2020.

[12] A. Venugopal, O. Yadukrishnan, and R. Nair T., "A svm based gender classification from children facial images using local binary and non-binary descriptors," in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 631–634, 2020.

[13] S. Mittal and V. S. Rajput, "Gender and age based census system for metropolitan cities," in *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pp. 1094–1097, 2020.

[14] A. Kuehlkamp and K. Bowyer, "Predicting gender from iris texture may be harder than it seems," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 904–912, 2019.

[15] O. Surinta and T. Khamket, "Gender recognition from facial images using local gradient feature descriptors," in *2019 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, pp. 1–6, 2019.

[16] P. Vashisth and K. Meehan, "Gender classification using twitter text data," in *2020 31st Irish Signals and Systems Conference (ISSC)*, pp. 1–6, 2020.

[17] T. Lekamge and T. Fernando, "Finding the gender of personal names and finding the effect of gana on personal names with long short term memory," in *2019 19th International Conference on Advances in ICT for Emerging Regions (ICTer)*, vol. 250, pp. 1–8, 2019.

[18] S. Mamgain, R. C Balabantaray, and A. K Das, "Author profiling: Prediction of gender and language variety from document," in *2019 International Conference on Information Technology (ICIT)*, pp. 473–477, 2019.

[19] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 2, pp. 604–624, 2021.

[20] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*, vol. 1. MIT press Cambridge, 2016.

[21] S. Amarappa and S. Sathyanarayana, "Kannada named entity recognition and classification (nerc) based on multinomial naïve bayes (mnb) classifier," *International Journal on Natural Language Computing*, vol. 4, 09 2015.

[22] L. P. Manik, A. F. Syafiandini, H. F. Mustika, Z. Akbar, and Y. Rianto, "Gender inference based on indonesian name and profile photo," in *2019 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, pp. 25–29, IEEE, 2019.

[23] S. Yuenyong and S. Sinthupinyo, "Gender classification of thai facebook usernames," *International Journal of Machine Learning and Computing*, vol. 10, no. 5, 2020.

[24] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[25] X. Zhang, J. Zhao, and Y. Lecun, "Character-level convolutional networks for text classification," *Advances in Neural Information Processing Systems*, vol. 2015, pp. 649–657, 2015.

[26] U. Kamath, J. Liu, and J. Whitaker, *Deep learning for NLP and speech recognition*, vol. 84. Springer, 2019.

[27] S. Haykin, *Neural Networks and Learning Machines, 3/E.* Pearson Education India, 2010.

[28] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder-decoder for statistical machine translation,” *arXiv preprint arXiv:1406.1078*, 2014.

[30] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” in *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.