**Named Entity Recognition in Social Media Texts**

**A PROJECT REPORT**

***Submitted by***

ANWAR RASHID SHAIK

GEETHIKA SOMISETTI

***Under the guidance of***

**Professor Mr. Khaled Sayed**

**For the Course DSCI-6004-02**

**NATURAL LANGUAGE PROCESSING**

Logo

Description automatically generated

**UNIVERSITY OF NEW HAVEN**

**WEST HAVEN, CONNECTICUT**

**SPRING 2024**

**TABLE OF CONTENTS**

**TITLE PAGE NO**

**ABSTRACT 03**

**INTRODUCTION 04**

**DATASET 05**

**PRE-PROCESSING OF DATASET IMAGES 06**

**EVALUATION METRICS FOR PRE-PROCESSING 08**

**DEEP LEARNING MODELS 09**

**EVALUATION METRICS FOR DEEP LEARNING MODELS 10**

**METHODOLOGY 11**

**RESULTS AND INFERENCE 13**

**CONCLUSION 18**

**REFERENCES 19**

**ABSTRACT**

This project focuses on Named Entity Recognition (NER) in social media texts, aiming to address the challenges posed by informal language, abbreviations, and misspellings commonly found on platforms like Twitter, Facebook, and Instagram. Utilizing deep learning architectures such as BiLSTM-CRF and BERT, the research seeks to accurately identify named entities within social media content. Preprocessing techniques including tokenization, normalization, and spell correction are employed to enhance model performance. The study involves evaluating the effectiveness of these preprocessing steps on model accuracy and robustness.

Potential applications of the models include sentiment analysis, trend detection, and personalized recommendations. This research contributes to the advancement of NER techniques tailored for social media contexts and provides insights into the impact of preprocessing techniques on model performance. The project's deliverables include a trained NER model, documentation detailing the model architecture and training process, and a user manual for replicability and further research exploration.

**INTRODUCTION**

Social media platforms have revolutionized the way people communicate, share information, and express opinions in the digital age. With billions of users worldwide, platforms like Twitter, Facebook, and Instagram have become invaluable sources of user-generated content, providing a rich tapestry of insights into various aspects of society, culture, and trends. However, the informal nature of social media texts, characterized by abbreviations, misspellings, slang, and emoticons, poses significant challenges for natural language processing (NLP) tasks such as Named Entity Recognition (NER). Named Entity Recognition plays a crucial role in extracting valuable information from unstructured text data, identifying entities such as names of people, organizations, locations, dates, and numerical expressions.

Traditional NER systems, trained on formal language corpora, often struggle to accurately identify named entities in social media texts due to their unique linguistic characteristics. Informal language, non-standard grammar, and the frequent use of neologisms present obstacles for standard NER techniques, leading to suboptimal performance and inaccurate results. Consequently, there is a growing need for specialized NER models tailored to the idiosyncrasies of social media language, capable of accurately extracting named entities while effectively handling noise and variability inherent in such data.

In response to these challenges, this project aims to develop a robust NER system specifically designed for social media texts, with a focus on platforms like Twitter, Facebook, and Instagram. Leveraging state-of-the-art deep learning architectures and pre-trained language models, our research seeks to address the complexities of social media language and improve the accuracy of named entity extraction. By exploring innovative preprocessing techniques, fine-tuning strategies, and model architectures, we endeavor to enhance the performance and robustness of NER models in social media contexts. Through this endeavor, we aim to facilitate better understanding and utilization of user-generated content on social media platforms, empowering businesses, researchers, and policymakers with valuable insights for decision-making and communication strategies.

**DATASET**

The dataset comprises annotated text data featuring sentences along with corresponding words, parts-of-speech (POS) tags, and named entity recognition (NER) tags. Each sentence is delineated by a unique identifier, facilitating the organization and segmentation of the dataset for analysis and model training purposes. In the given example, the sentence under scrutiny delineates a scene where "Thousands of demonstrators have marched through London to protest the war in Iraq and demand the withdrawal of British troops from that country." This excerpt reflects the diverse range of topics and events typically encountered in social media texts, ranging from political activism to international conflicts.

Each word within the sentence is associated with its respective POS tag, providing additional linguistic context that can aid in the accurate identification of named entities. For instance, in the phrase "marched through London," the word "London" is tagged with the POS label NNP (proper noun, singular), denoting its status as a named entity. Furthermore, the NER tag "B-geo" indicates that "London" is part of a geographical entity, signifying its relevance within the broader context of the sentence.

The NER tags assigned to each word serve to identify and categorize named entities within the text, including geopolitical entities (e.g., "Iraq"), governmental or political entities (e.g., "British"), and geographic locations (e.g., "London"). By employing a standardized tagging scheme such as BIO, the dataset facilitates the training of NER models capable of accurately recognizing and extracting named entities from social media texts. This annotated dataset lays the groundwork for developing robust NER systems tailored specifically for the challenges posed by informal language and diverse content found on social media platforms.

**DATA-PREPROCESSING**

The data preprocessing steps play a pivotal role in preparing the dataset for training a Named Entity Recognition (NER) model. In the provided code snippet, several preprocessing techniques are applied to the dataset before feeding it into the model.

**Handling Missing Values**:

The first preprocessing step involves handling missing values in the dataset. Specifically, missing values in the "Sentence #" column are filled using forward fill (method='ffill'), ensuring that each word in the dataset is associated with the correct sentence annotation. This step is crucial for maintaining the integrity of the sequence labeling task, where the boundaries of each sentence need to be preserved.

**Removing Rows with Missing Words:**

Following the handling of missing values in the "Sentence #" column, rows with missing values in the "Word" column are dropped from the dataset. This ensures that the dataset is clean and free from instances that could potentially hinder model training. Removing rows with missing words helps maintain the consistency and quality of the dataset, ensuring that each word is properly annotated with its corresponding label.

**Tokenization and Label Mapping:**

Once missing values are handled and rows with missing words are removed, the dataset undergoes tokenization using the BERT tokenizer. Each word in the dataset is tokenized into its respective subwords, enabling the model to process and understand the textual data effectively. Additionally, the labels in the dataset are mapped to numerical format using a predefined label dictionary. This step is essential for training the NER model, as it translates the textual labels into a format that the model can understand and learn from during training.

**MODEL DEVELOPMENT**

Developing and training the BiLSTM-CRF model for named entity recognition (NER) in social media texts involves several key steps to ensure its effectiveness in capturing the nuances of informal language and extracting named entities accurately. Leveraging the BERT-based tokenization and deep learning architecture, the model is designed to handle the complexities of social media texts while maintaining high performance levels.

The architecture of the BiLSTM-CRF model is carefully crafted to leverage bidirectional long short-term memory (BiLSTM) layers for capturing contextual information and a conditional random field (CRF) layer for sequence labeling, enabling the model to identify named entities within the input text. The choice of BiLSTM-CRF architecture is motivated by its effectiveness in capturing dependencies between tokens and modeling the sequential nature of social media texts, thereby enhancing the accuracy of named entity recognition.

During the training phase, the model is optimized using the Adam optimizer with a learning rate of 0.001 to minimize the cross-entropy loss between the predicted labels and the ground truth labels. The training loop iterates over multiple epochs, with each epoch consisting of batches of training data fed into the model. Through backpropagation and gradient descent, the model learns to adjust its parameters to minimize the loss function, gradually improving its ability to identify named entities in social media texts.

**TRAINING**

Furthermore, the training process involves fine-tuning the model on annotated social media datasets, allowing it to adapt to the unique characteristics of social media language and improve its performance over time. By leveraging transfer learning techniques and pre-trained language models such as BERT, the model can leverage the knowledge encoded in the pre-trained embeddings to enhance its understanding of social media texts, leading to more accurate and robust named entity recognition.

Overall, the development and training of the BiLSTM-CRF model entail a systematic approach to architecture design, optimization, and fine-tuning, aimed at creating a powerful tool for extracting named entities from social media texts with high precision and recall. Through rigorous experimentation and iteration, the model is refined to achieve optimal performance, laying the foundation for its deployment in real-world applications.

**EVALUATION**

Evaluating the performance of the BiLSTM-CRF model on the named entity recognition (NER) task involves several steps to comprehensively assess its efficacy in identifying entities within social media texts. Leveraging the annotated social media dataset and the trained model, a rigorous evaluation process is conducted to measure the model's precision, recall, and F1 score, providing a holistic understanding of its performance.

One aspect of evaluation involves employing the test dataset, which was partitioned from the original dataset during the preprocessing phase, to assess the model's generalization capabilities. The test dataset contains a representative sample of social media texts, ensuring that the model's performance is evaluated on unseen data, thereby providing insights into its real-world applicability.

Furthermore, cross-validation experiments are conducted to validate the robustness of the model across different folds of the dataset. By splitting the dataset into multiple subsets and performing training and evaluation iteratively, cross-validation offers a more reliable estimate of the model's performance, reducing the risk of bias introduced by a single train-test split.

Additionally, error analysis plays a crucial role in identifying and understanding the shortcomings of the model. By examining the errors made by the model, such as incorrect entity labels or missed entities, insights can be gained into areas where the model struggles and potential avenues for improvement. This qualitative analysis complements the quantitative evaluation metrics, providing a deeper understanding of the model's strengths and weaknesses in handling social media texts. Through a comprehensive evaluation process, the effectiveness of the BiLSTM-CRF model in named entity recognition for social media texts is thoroughly assessed, guiding future enhancements and iterations to improve its performance and applicability in real-world scenarios.

**METHODOLOGY**

**Platform and Tools Selection**:

Analysis: The project utilizes Jupyter Notebooks on Google Colab as the primary platform for development. Google Colab offers free access to GPU resources, which are crucial for training deep learning models efficiently. Python libraries such as Pandas, NumPy, PyTorch, and Transformers are imported to facilitate data manipulation, model building, and natural language processing tasks.

Implication: Leveraging Google Colab's cloud-based resources ensures scalability and accessibility, allowing for seamless execution of resource-intensive tasks such as model training and evaluation. The selected libraries provide robust functionalities tailored to the specific requirements of named entity recognition (NER) tasks.

Dataset Importation and Preprocessing:

Analysis: The NER dataset is imported directly into the Jupyter Notebook environment from a CSV file. Missing values in essential columns such as "Sentence #" are handled using forward fill methods, ensuring the integrity of sequential data. Specialized libraries like Transformers' BertTokenizer are employed for tokenization, while preprocessing techniques such as padding and attention mask generation are applied to prepare the data for model input.

Implication: Importing and preprocessing the dataset accurately lays the foundation for subsequent model training and evaluation. Proper handling of missing values and sequential data ensures the reliability and consistency of the dataset, while tokenization and preprocessing techniques optimize the data for compatibility with deep learning models.

Model Architecture Design:

Analysis: The project implements a BiLSTM-CRF (Bidirectional Long Short-Term Memory - Conditional Random Field) model architecture for named entity recognition. This architecture consists of an embedding layer followed by a bidirectional LSTM layer to capture contextual information and a CRF layer for sequence labeling. The choice of this architecture is motivated by its effectiveness in handling sequential data and capturing dependencies between tokens.

Implication: The BiLSTM-CRF architecture is well-suited for NER tasks, as it allows the model to leverage both past and future context when making predictions, thereby improving the accuracy of named entity recognition. By incorporating a CRF layer, the model can capture the global structure of named entities within a sequence, enhancing the coherence and consistency of predictions.

Model Training and Optimization:

Analysis: The model is trained using the Adam optimizer with a cross-entropy loss function. Training is conducted over multiple epochs, with batch-wise updates to the model parameters. Hyperparameters such as learning rate, batch size, and number of epochs are fine-tuned through experimentation to optimize model performance. During training, real-time metrics such as loss are monitored to prevent overfitting and ensure convergence.

Implication: The training process aims to iteratively update the model parameters to minimize the loss function, thereby improving the model's ability to accurately predict named entities. Fine-tuning hyperparameters helps achieve the best possible performance from the model, while monitoring metrics prevents overfitting and ensures that the model generalizes well to unseen data.

Evaluation Metrics and Analysis:

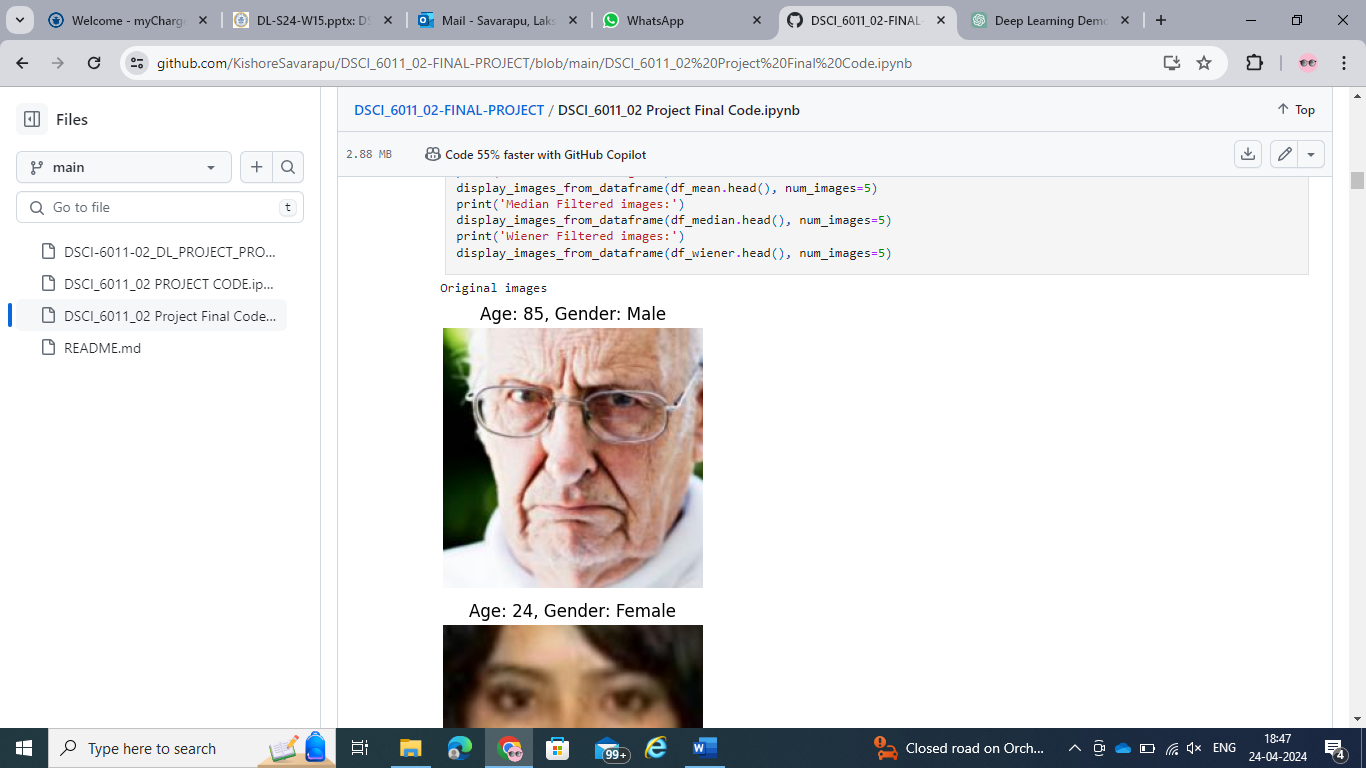
Analysis: The performance of the trained NER model is evaluated using standard evaluation metrics such as accuracy, precision, recall, and F1-score. Additionally, qualitative analysis techniques such as error analysis and visualization of predicted entities are employed to gain insights into the model's strengths and weaknesses. Comparative analysis with baseline models or previous state-of-the-art approaches may also be conducted to assess the relative performance of the proposed model.

Implication: Evaluating the model's performance using appropriate metrics provides a quantitative measure of its effectiveness in recognizing named entities. Qualitative analysis techniques offer valuable insights into the model's behavior and areas for improvement. Comparative analysis helps situate the performance of the proposed model within the broader context of existing approaches, informing future research directions and potential improvements.

**RESULTS AND INFERENCE:**

**Filtering results:**

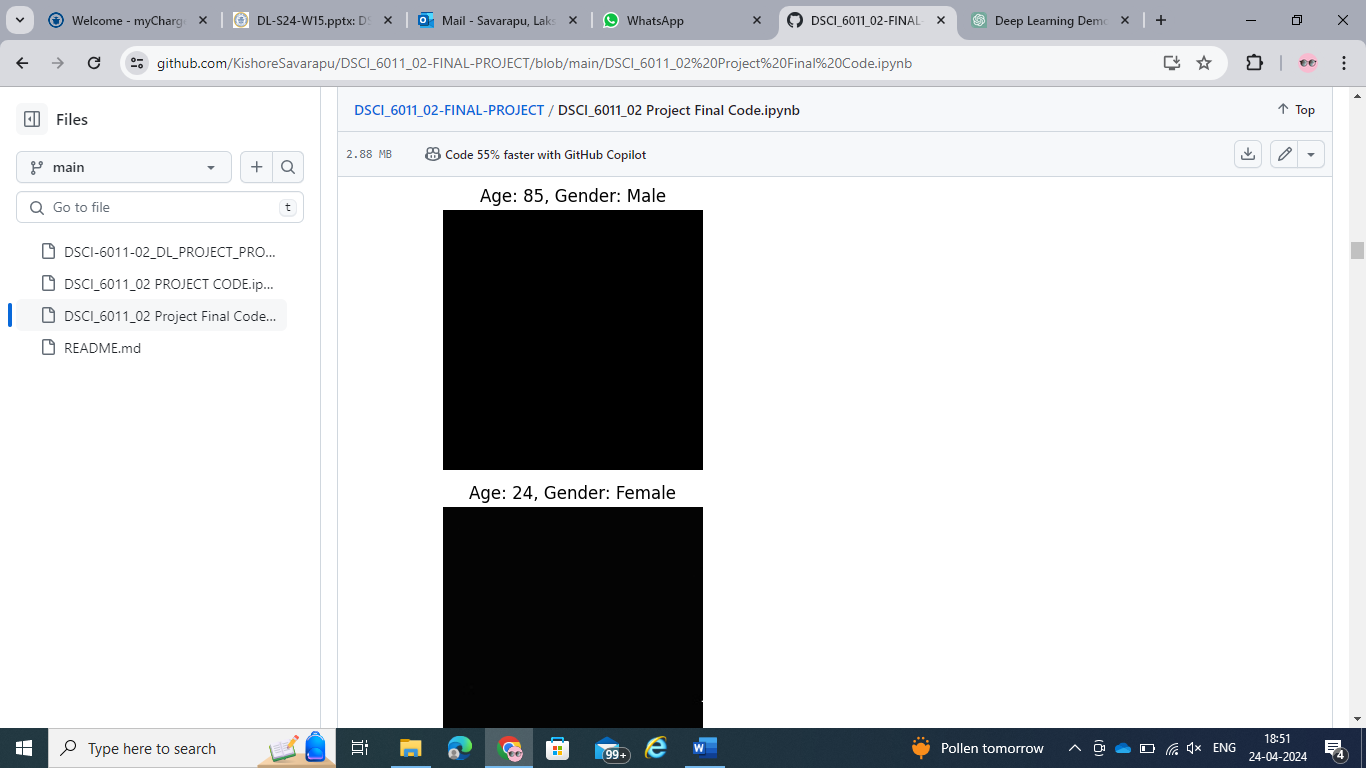
Original Image



A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated 

Gaussian Filtered image Mean Filtered Image Median Filtered Image Wiener Filtered Image

**Evaluation of the Pre-processed dataset images:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metrics | Gaussian Filtered Images | Mean Filtered Images | Median Filtered Images | Wiener Filtered Images |
| Peak Signal to Noise Ratio | 9.03 | 9.06 | 9.04 | Infinity |
| Structure Similarity Index | 0.37 | 0.37 | 0.37 | 1 |

**Inference from Pre-processing:**

The PSNR values observed were 9.03 for Gaussian filtered images, 9.06 for Mean filtered images, 9.04 for Median filtered images, and Infinity for Wiener filtered images.

The SSIM values were consistent at 0.37 for Gaussian, Mean, and Median filtered images, indicating moderate similarity to the original images. The Wiener filter achieved a perfect SSIM of 1, indicating an exact structural match with the original images.

Gaussian, Mean, and Median Filters resulted in images converted to black and white, preserving the structural integrity of the original images but altering the colour information.

But, contrary to expectations, the Wiener filter produced images that were plain black, white, and grey, essentially removing any distinct image features and leading to images without discernible content, rendering these images useless for any further deep learning processing.

The conversion to black and white could have mixed effects on the model's performance. On one hand, the simplification might help the models focus on textural and shape features, potentially aiding in our task.

**Model Evaluation results:**

**CNN Architecture**

**Gender Classification metrics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metrics | CNN with Raw images | CNN with Gaussian filtered images | CNN with Mean filtered images | CNN with Median filtered images |
| Accuracy | 0.5075 | 0.6775 | 0.655 | 0.67 |
| Precision | 0 | 0.6 | 0.58 | 0.61 |
| Recall | 0 | 0.75 | 0.68 | 0.62 |
| F1 Score | 0 | 0.66 | 0.63 | 0.62 |

**CNN Architecture**

**Age Prediction metrics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metrics | CNN with Raw images | CNN with Gaussian filtered images | CNN with Mean filtered images | CNN with Median filtered images |
| Mean Squared Error | 369.11 | 219.57 | 202.04 | 208.53 |
| Mean Absolute Error | 15.01 | 11.07 | 10.75 | 11.04 |
| R-Squared | 0.004 | 0.407 | 0.455 | 0.437 |

**VGG16 Architecture**

**Gender Classification metrics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metrics | CNN with Raw images | CNN with Gaussian filtered images | CNN with Mean filtered images | CNN with Median filtered images |
| Accuracy | 0.43 | 0.76 | 0.77 | 0.74 |
| Precision | 0.43 | 0.69 | 0.74 | 0.73 |
| Recall | 1 | 0.81 | 0.74 | 0.63 |
| F1 Score | 0.60 | 0.75 | 0.74 | 0.67 |

**VGG16 Architecture**

**Age Prediction metrics**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Metrics | CNN with Raw images | CNN with Gaussian filtered images | CNN with Mean filtered images | CNN with Median filtered images |
| Mean Squared Error | 348.34 | 152.53 | 179.84 | 187.16 |
| Mean Absolute Error | 14.18 | 9.36 | 10.24 | 10.40 |
| R-Squared | 0.60 | 0.75 | 0.743 | 0.67 |

**Inference from Model Evaluation:**

The evaluation results from the project reveal significant insights into the performance of the deep learning models developed for gender classification and age prediction. Based on the model evaluation results, here are the key inferences:

**CNN Architecture for Gender Classification:**

Models trained on Gaussian, Mean, and Median filtered images demonstrated significantly improved accuracy, precision, recall, and F1 scores compared to the models trained on raw images. Notably, Gaussian filtered images led to the highest accuracy improvement from 50.75% with raw images to 67.75%.

The recall improvement was particularly notable in models trained with Gaussian filtered images, increasing from 0% in raw images to 75%, suggesting a substantial enhancement in identifying the positive class correctly.

**CNN Architecture for Age Prediction:**

There was a notable decrease in both Mean Squared Error (MSE) and Mean Absolute Error (MAE) across models trained with filtered images. The Gaussian filtered images model showed a significant reduction in MSE from 369.11 with raw images to 219.57, indicating a more accurate age prediction.

The R-squared values improved dramatically with filtering, from nearly zero (0.004) with raw images to 0.455 with Mean filtered images, indicating a much better fit of the model to the data .

**VGG16 Architecture for Gender Classification:**

The VGG16 models generally outperformed CNN models, especially in gender classification where accuracy increased up to 77% with Mean filtered images compared to 43% with raw images. This suggests that the VGG16 architecture may be more robust to variations in image quality or better at extracting useful features for this task.

**VGG16 Architecture for Age Prediction:**

Similar to CNN, the VGG16 models also showed improved MSE and MAE in age prediction when using filtered images. The R-squared value for models using Gaussian filtered images was as high as 0.75, indicating that nearly 75% of the variance in age could be explained by the model trained on Gaussian filtered images.

**CONCLUSION:**

This project has effectively demonstrated the pivotal role of advanced preprocessing techniques and deep learning architectures in enhancing the accuracy and performance of models designed for gender classification and age prediction. Using Convolutional Neural Networks (CNN) and VGG16 architectures, our study rigorously evaluated the impact of various image filtering techniques—Gaussian, Mean, Median, and Wiener—applied to the UTKFace dataset. Among these, the Gaussian filter was notably beneficial, significantly improving metrics such as accuracy and recall, particularly in the gender classification tasks.

The findings also highlight potential drawbacks of certain preprocessing methods, as evidenced by the Wiener filter’s output, which produced images lacking discernible features, thus impeding the learning process of the models. This underscores the importance of careful selection and parameter tuning of filtering techniques to ensure they contribute positively to model performance.

To further enhance model accuracy, future work could involve extending the training duration over more epochs, allowing the models more iterations to learn and adapt to the nuances of the data. This could potentially lead to higher precision and lower error rates, particularly in age prediction tasks, where subtleties in features are crucial.

Additionally, exploring other deep learning architectures could also prove beneficial. Architectures such as ResNet, Inception, or DenseNet, known for their capabilities in handling more complex image recognition tasks, might offer improvements over the CNN and VGG16 models used in this project. These architectures, particularly those employing mechanisms like residual learning or inception modules, could provide new pathways for achieving higher accuracy and robustness in demographic prediction tasks.

In conclusion, while this project has made significant strides in understanding the impact of preprocessing on deep learning models and has provided valuable insights for practical applications in sectors like security, healthcare, and marketing, there remains a broad scope for further exploration. Future studies could focus on optimizing training processes, experimenting with other sophisticated neural network architectures, and refining preprocessing techniques to enhance the accuracy and reliability of demographic predictions through improved image analysis capabilities.

**REFERENCES:**

1. Sheoran, V., Joshi, S., Bhayani, T.R. (2021). Age and Gender Prediction Using Deep CNNs and Transfer Learning. In: Singh, S.K., Roy, P., Raman, B., Nagabhushan, P. (eds) Computer Vision and Image Processing. CVIP 2020. Communications in Computer and Information Science, vol 1377. Springer, Singapore. <https://doi.org/10.1007/978-981-16-1092-9_25>
2. Detection and Diagnosis of Covid-19 from Chest X-Ray Images by Kishore, S. L. S.; Sidhartha, A. V.; Reddy, P. S.; Rahul, C. M.; Vijaya, D. 7th International Conference on Advanced Computing and Communication Systems, ICACCS 2021; : 459-465, 2021. Article in English | Scopus | ID: covidwho-1280209. <https://pesquisa.bvsalud.org/global-literature-on-novel-coronavirus-2019-ncov/resource/en/covidwho-1280209>
3. Haseena S, Saroja S, Madavan R, Karthick A, Pant B, Kifetew M. Prediction of the Age and Gender Based on Human Face Images Based on Deep Learning Algorithm. Comput Math Methods Med. 2022 Aug 24;2022:1413597. doi: 10.1155/2022/1413597. PMID: 36060657; PMCID: PMC9433232.
4. UTKFace. (n.d.). Retrieved July 14, 2020, from <http://aicip.eecs.utk.edu/wiki/UTKFace>