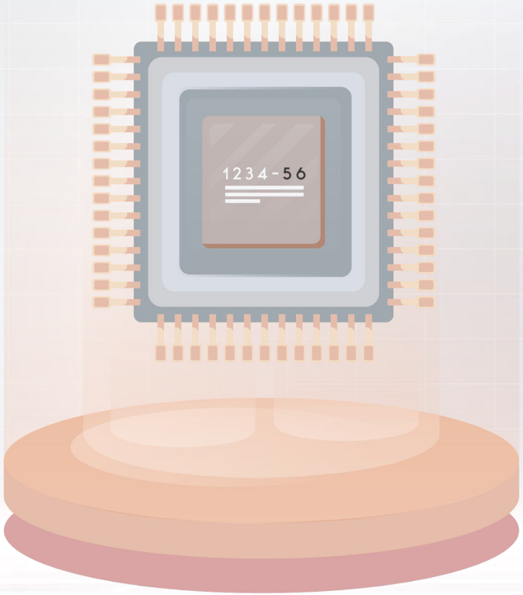
Milestone II

Arabic Language Classification

By:
Ahmed Jaheen
Aedan Ounsamone

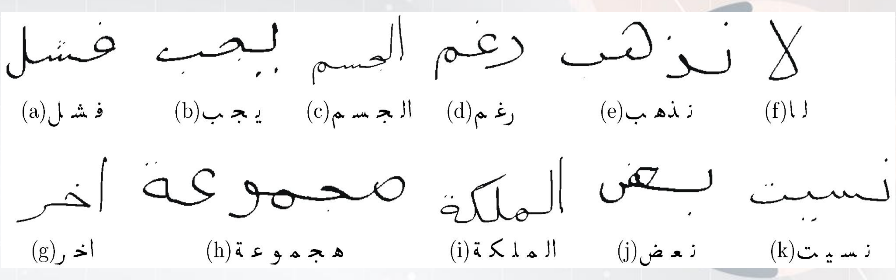
TABLE OF CONTENTS



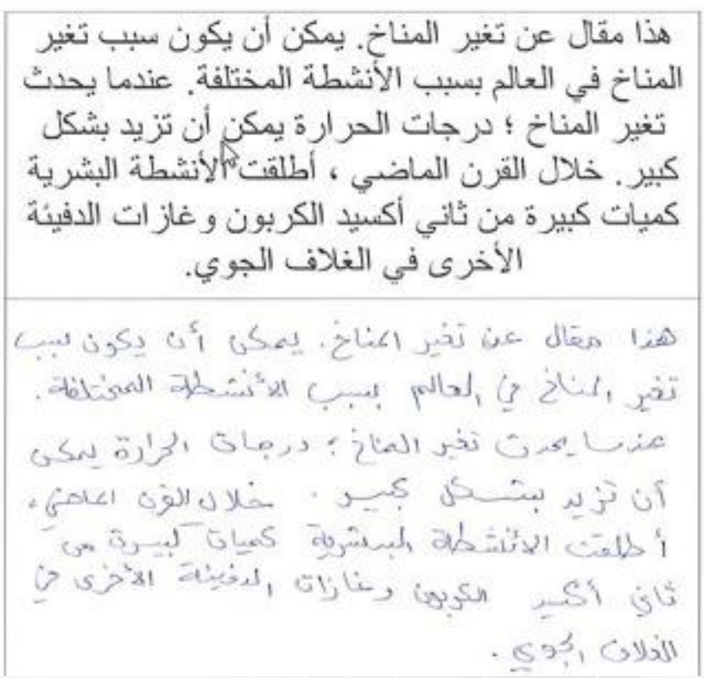
01

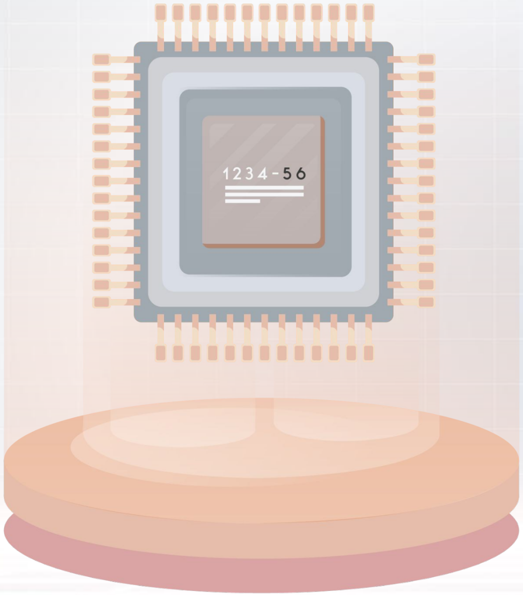
Problem Statement

Problem statement



Arabic handwriting recognition presents unique challenges due to its cursive nature and character variability, which have hindered the development of robust recognition models. Our project aims to address these challenges by evaluating existing models on datasets and proposing enhancements to improve accuracy and efficiency in Arabic text recognition systems.

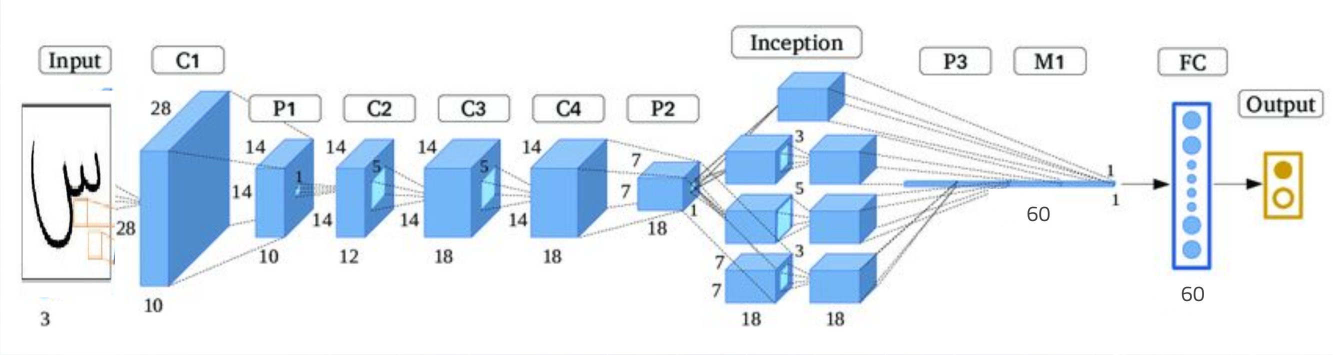

هذا مقال عن تغير المناخ. يمكن أن يكون سبب تغير المناخ في العالم بسبب الأنشطة المختلفة. عندما يحدث تغير المناخ ؛ درجات الحرارة يمكن أن تزيد بشكل كبير. خلال القرن الماضي ، أطلقت الأنشطة البشرية كميات كبيرة من ثاني أكسيد الكربون وغازات الدفيئة الأخرى في الغلاف الجوي.



02

Original Model

Original Model



The GoogleNet model was chosen for its alignment with our research direction of incorporating an noisy label mechanisms and vision transformers mechanisms for arabic letter classification. It has straightforward architecture for easier understanding and updates, user-friendly implementation, and efficiency in training. Also, it includes minimal file count and use of recent frameworks, and It got the highest accuracy in past papers.

03

Proposed Updates and Solutions

Proposed Updates and Solutions

- Use Regularization Techniques, a Learning Rate Scheduler, and data augmentation techniques (DONE in Milestone I).
- Test different pre-trained models & evaluate accuracies on data (DONE in Milestone I).
- Fine-tuning best pre-trained model (DONE in Milestone I).
- Try to apply the proposed noisy labels handling mechanism to see its effect on the model (Milestone II).
- Use vision transformers model and trying different attention mechanisms (Milestone II & Final).
- Use Adaboost & different ensemble classifiers (Final).
- Using OCR Technique to validate whole words (Final).

04

Progress done
&
results.

Milestone I:

1. Successful deployment of the baseline model on AHAWP dataset.
2. Applying Data augmentation on the AHAWP dataset and retraining the model.
3. Finding the false Predictions and make research on Noisy labels
4. Running the model on the AHAWP dataset with L2 and Dropout regularizers.
5. Researching noisy labels handling mechanisms

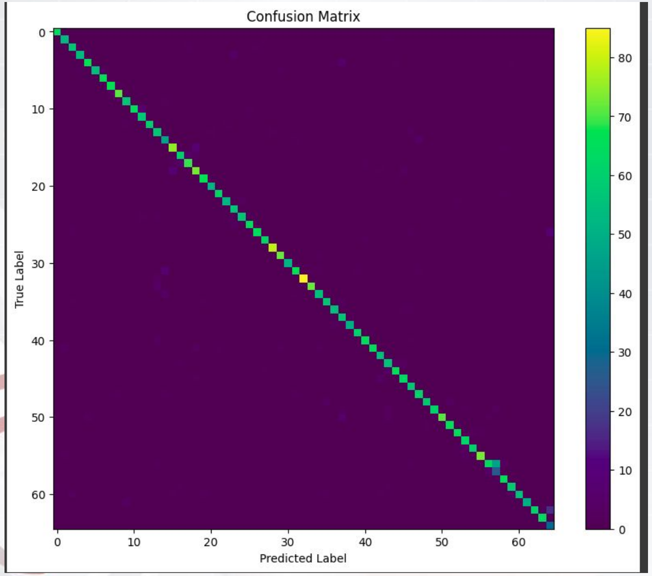
Milestone II:

1. Applying label smoothing and the focal loss as mechanisms of handling the issue of noisy labels.
2. After researching JNPL noisy label, we deployed it in an optimistic approach.
3. Applying region based attention and self attention as attention mechanism in the model.
4. Researching OCR techniques to detect words.

Results of deploying the original model on AHAWP

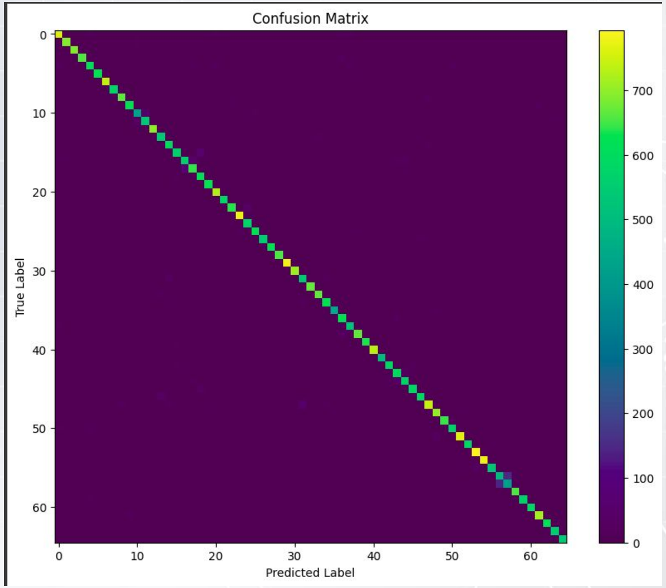
1) Without Updates:

- Accuracy= 0.9256
- Confusion Matrix:



2) With data augmentation and regularization:

- Accuracy= 0.9481
- Confusion Matrix:



The results of Milestone II.

First, Usage of Label Smoothing (Research)

- Prevents Overconfidence and Overfitting: Label Smoothing Loss stops the model from being overly confident and overfitting the training data.

- Enhances Generalization: Encourages the model to generalize better to unseen data.
- Introduces Uncertainty: Adds a controlled level of uncertainty during training to mitigate overfitting.

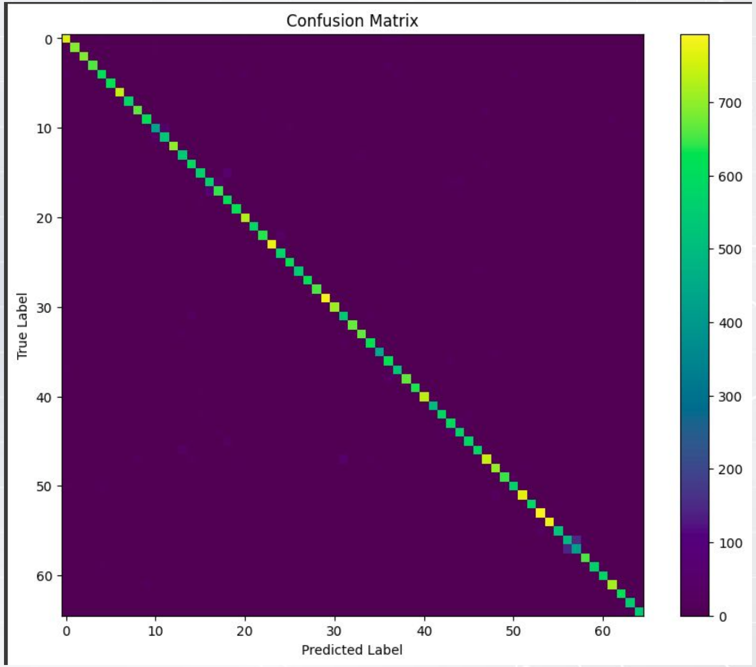
- Discourages Extreme Confidence: Prevents the model from being too certain in its predictions, promoting a more balanced distribution of probabilities.

- Smoothing Factor: Regulates the intensity of regularization. Higher values result in more uniform probabilities and greater regularization.

Usage of Label Smoothing (Results)

- Training Accuracy= 0.9753
- Validation Accuracy= 0.9431
- Testing Accuracy= 0.9476
- Weighted Precision= 0.9412
- Unweighted Precision = 0.9405
- Weighted Recall = 0.9417
- Unweighted Recall=0.9401
- F-1 Score=0.9415

- Confusion Matrix:



Findings after Label Smoothing.

1. Usage of label smoothing did not help in improving the model performance.
2. The accuracy was lower.
3. The model was able to predict all of the classes.

Hence, We are going to continue experimenting with different noisy labels handling mechanisms.

Second, Usage of Focal Loss (Research)

- Targets Imbalanced Data: Focal Loss is designed to address class imbalance by focusing training on hard-to-classify examples.

- Reduces Overfitting to Easy Examples: Helps prevent the model from overfitting to easy examples by reducing the loss contribution from easily classified instances.

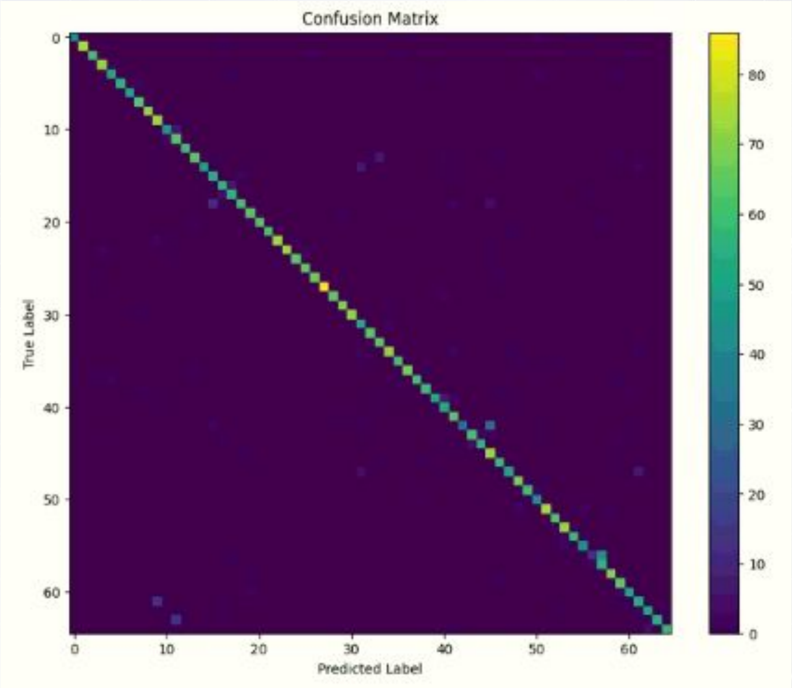
- Enhances Attention to Difficult Cases: Increases the model's attention to more challenging cases, which are often underrepresented or harder to predict correctly.

- Adjustable Focus Parameter: Incorporates a focusing parameter that adjusts the emphasis on hard examples, allowing for customizable sensitivity to difficult classifications.

Usage of Focal Loss (Results)

- Training Accuracy= 0.9797
- Validation Accuracy= 0.9215
- Testing Accuracy= 0.9182
- Weighted Precision= 0.9312
- Unweighted Precision = 0.9288
- Weighted Recall = 0.9215
- Unweighted Recall=0.9205
- F-1 Score=0.9196

- Confusion Matrix:



Findings after Focal Loss.

1. Bad performance than label smoothing, which did not help in improving the model performance.
2. The model was able to predict all of the classes.
3. The accuracy was very low (less than original model by 4%).

Hence, we are going to continue experimenting with JNPL noisy label handling mechanism.

Third, Usage of JNPL (Research)

- It’s specifically designed to identify and handle noisy labels through a joint learning process that involves filtering noisy samples (negative learning) and improving the model's performance with the cleaned data (positive learning).

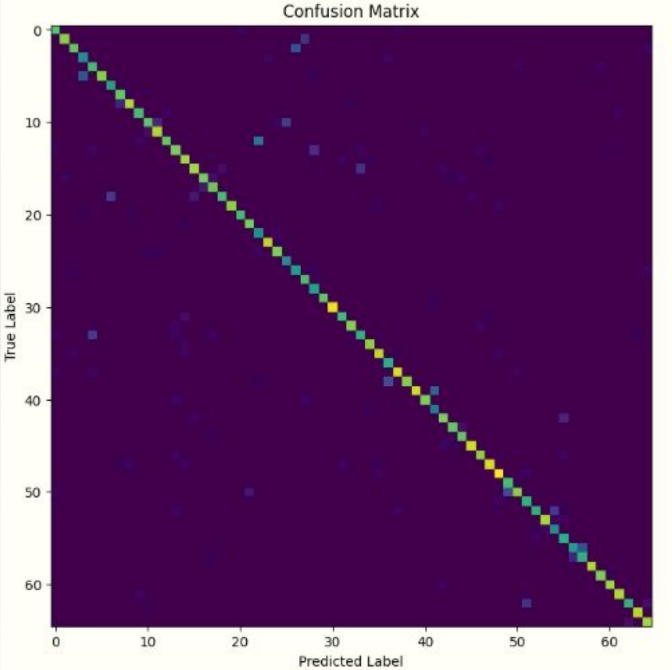
- It uses a unified single-stage pipeline, and incorporating pseudo-labeling to make better use of the noisy data.

Link to the paper:
https://openaccess.thecvf.com/content/CVPR2021/papers/Kim_Joint_Negative_and_Positive_Learning_for_Noisy_Labels_CVPR_2021_paper.pdf

Usage of JNPL (Results)

- Training Accuracy= 0.9990
- Validation Accuracy= 0.9602
- Testing Accuracy= 0.9801
- Weighted Precision= 0.9804
- Unweighted Precision = 0.9780
- Weighted Recall = 0.9638
- Unweighted Recall=0.9634
- F-1 Score=0.9633

- Confusion Matrix:



Findings after JNPL.

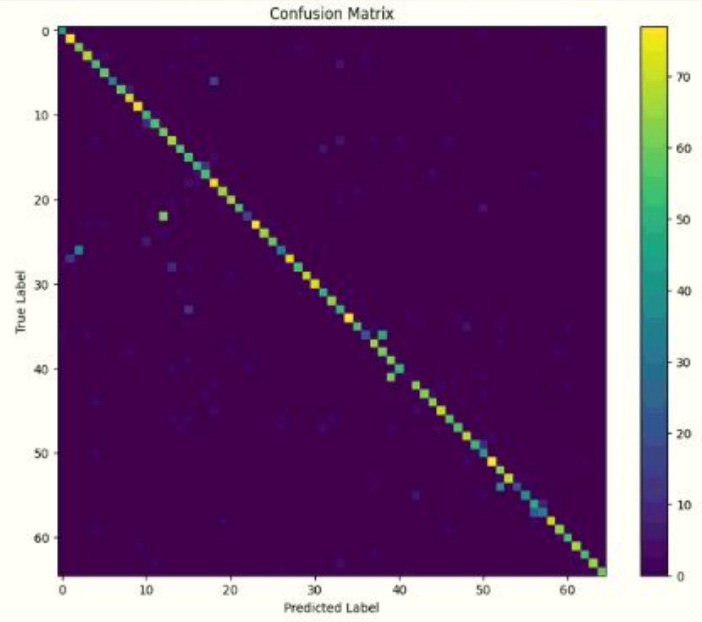
1. The performance of JNPL was very good comparing to label smoothing and focal loss which helped in improving the model performance.
2. The model was able to predict all of the classes.
3. The accuracy was higher than the original model by 3%.

Now, we are going to use different attention mechanisms to improve the model further.

Fourth, Usage of Region based attention. (Results)

- Training Accuracy= 0.8411
- Validation Accuracy= 0.8737
- Testing Accuracy= 0.8860
- Weighted Precision= 0.8974
- Unweighted Precision = 0.8938
- Weighted Recall = 0.8737
- Unweighted Recall=0.8748
- F-1 Score=0.8649

- Confusion Matrix:



Findings after Region based attention.

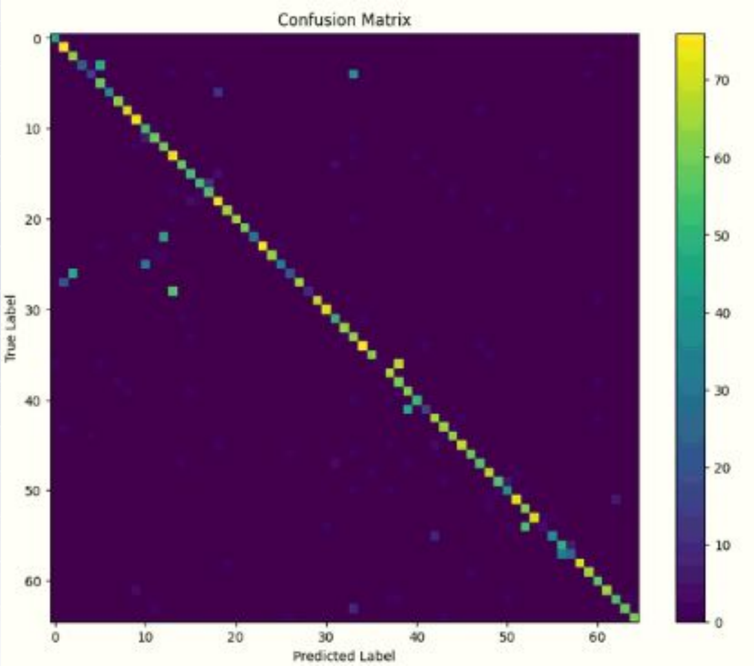
1. Usage of region based attention did not help in improving the model performance.
2. It increased the complexity of the model (we might change the hyperparameters and check again)
3. The accuracy was very lower than the original model.

Hence, we are going to use different attention mechanisms.

Fifth, Usage of self attention. (Results)

- Training Accuracy= 0.8398
- Validation Accuracy= 0.8397
- Testing Accuracy= 0.8598
- Weighted Precision= 0.8854
- Unweighted Precision = 0.8814
- Weighted Recall = 0.8397
- Unweighted Recall=0.8414
- F-1 Score=0.8217

- Confusion Matrix:



Findings after self attention.

1. Usage of self attention did not help in improving the model performance.
2. It also increased the complexity of the model.
3. The accuracy was lower than the original model.

Hence, we are going to use different attention mechanisms and vision transformers and ensemble classifier in final milestone.

Evaluation of updated model vs original Model

- Application of regularization reduced the issue of overfitting that the model suffered from and increased the overall accuracy (Done in Milestone I).
- Application of data augmentation made the model capable of recognising all classes and not leaving focus on five important classes (Done in Milestone I).
- Application of JNPL (noisy label handling mechanism) did improve the performance (Done in Milestone II).
- Application of different attention mechanisms did not improve the performance (Done in Milestone II).

05

Updates & Steps to Final Milestone

Updates & Steps to Final Milestone

1. Use Different vision transformers and trying different attention mechanisms.
2. Use Adaboost & different ensemble classifiers.
3. Using OCR Technique to validate whole words.

06

Members' contributions.

Members' contributions.

Jaheen

- Finding and trying multiple noisy labels handling techniques.
- Applying Label smoothing to AHAWP dataset.
- Applying JNPL to AHAWP dataset.
- Experimented with Region based attention.
- Researching for OCR Techniques.

Aedan

- Experimented with Region based attention.
- Applying Focal Loss to AHAWP dataset.

Thanks!