50.035 Computer Vision

Research Final Project

# Deep Learning Lip Reading Model

Group 11

Kum Yu Rong

Chen Yu Tung

Wongsaphat P.

John-David Tan

# Table of Contents

# I. Introduction

Lip reading involves interpreting spoken language by analysing the movements of a speaker's lips and face, traditionally used by individuals with hearing impairments. Recent advances in AI and computer vision have enabled the development of automated lip-reading systems, which convert visual speech into text, offering new possibilities for enhancing communication.

In this study, we are exploring the potential of artificial intelligence to revolutionize the field of lip-reading. Our objective is to improve the capability of visual speech to text conversion in AI-driven systems. This entails a thorough examination of the lip-reading models that are currently in use, the application of different algorithms, and extensive testing through a number of experiments intended to evaluate and improve their efficacy. Through a comprehensive assessment of previous research, we will begin with our project by researching the state of automated lip-reading today. The phan involves reviewing a variety of models that have been extensively discussed in scholarly works, comprehending their fundamental workings, and identifying potential areas for improvement.

Beyond merely replicating existing results, we hope to improve these models by changing up ways to simplify the model and generating new insights. We plan to improve these systems through research, pushing the limits of what is currently possible with automated lip-reading tools. Our project's goal is to push lip-reading systems' capabilities.

# II. Background

The accuracy and viability of automated lip-reading systems have greatly improved recently thanks to developments in computer vision and deep learning, making them useful for applications in speech recognition, security, and voice augmentation. Two research publications, each concentrating on a distinct facet of automated lip-reading and its applications, provide insights into the most recent techniques and difficulties in this topic.

In order to create a system for lip-reading in Turkish, "Turkish Lip-Reading Using Bi-LSTM and Deep Learning Models" (2022) used CNNs with Bidirectional Long Short-Term Memory (Bi-LSTM) networks for spoken word and sentence classification. The researchers built two new datasets to cover this gap: one for word recognition, with 111 classes, and another for sentence recognition, with 113 classes. This was necessary because the majority of existing lip-reading datasets are language-specific and greatly differ in size. The video footage was gathered with constant lighting and camera settings in order to minimise variability.

The Turkish lip-reading system consisted of three main stages. Using the MediaPipe framework, lip areas in every video frame were identified in the initial stage of lip detection and preprocessing. Preprocessing operations like rotating and resizing the identified regions were performed after this to guarantee consistency across the dataset. Using CNNs, a model similar to ResNet-18 was used in the second stage of feature extraction to extract spatial information on the shape and movement of the lips from the photos. In the third and last step, temporal modelling was carried out using Bi-LSTM networks, which used the retrieved features to model the lip movements' temporal dynamics and categorise the input as a particular word or sentence.The researchers discovered that their best-performing model attained an accuracy of 88.55% for phrase identification and 84.5% for word recognition. This showed the efficacy of merging Bi-LSTM networks for temporal modelling with CNNs for spatial feature extraction.

The study "Lip-Reading Driven Deep Learning Approach for Speech Enhancement" (2021) investigated the use of lip-reading to enhance speech intelligibility and quality in noisy settings. To improve voice comprehension in such circumstances, the scientists suggested a novel deep learning framework that combined audio signal processing techniques with visual speech information. The strategy was divided into two main stages. A stacked Long Short-Term Memory (LSTM) network was utilised in the first stage of the lip-reading regression model to estimate clean audio features based on the visual input of lip movements. In the second stage, the noisy audio was subjected to an Enhanced Visually Derived Wiener Filtering (EVWF) procedure, which used the visual data from lip-reading to overcome the drawbacks of conventional Wiener filtering and produce a clean audio power spectrum. This approach was successful in outperforming more conventional methods such as spectral subtraction and minimum mean square error (MMSE) filtering.

Even while these research' findings were promising, they also pointed to several possible areas for development. Including sophisticated temporal models is one recommendation for improving performance. Although the research employed LSTM and Bi-LSTM networks for temporal modelling, more complex structures, such transformers, might enhance performance even further. Using attention techniques to focus on significant temporal patterns and capturing long-range interdependence, transformers have demonstrated success in various sequential tasks. In order to improve lip reading accuracy, future studies may investigate the incorporation of transformers.

In our exploration of automated lip reading research papers, we previously examined models that aim to improve speech intelligence in noisy settings. While these studies provided significant insights into the application of automated lip reading technologies, we have decided to reference and utilise LipNet model as shown in the research paper "LipNet: End-to-End Sentence-level Lipreading" by Yannis M. Assael et al., because of its creative approach and outstanding performance metrics.

Instead of just identifying words or phrases, LipNet sets itself apart from the other research models as the first end-to-end model to be able to predict sentence-level sequences. The Turkish lip-reading model utilises CNN with Bi-LSTM networks and the LipNet model incorporated and combined connectionist temporal features (CTC), recurrent neural networks (RNNs) ,spatiotemporal convolutional neural networks (STCNNs) and Bidirectional Gated Recurrent Unit (Bi-GRU) to successfully integrate spatial characteristics which the Turkish model is unable to achieve. The LipNet model also has empirical data that shows that it outperformed both human lip readings and previously established world-level benchmarks by reaching 95.2% accuracy in sentence-level word recognition on the GRID rubric. After learning that LipNet can perform better than other models that mostly just rely on temporal modeling, we believe that this is the ideal model that perfectly aligns with the objectives of our study.

|  | LipNet Model | Turkish Lip-Reading Model | General Speech Enhancement Models |
|---|---|---|---|
| Model Architecture | Uses STCNNs and GRUs networks | Uses CNNs and Bi-LSTM networks | Often use LSTM and other RNN networks |
| Output Level | Designed for sentence-level prediction, handling sequences directly and outputting full textual sentences. | Focuses on word and sentence classification, typically requiring post-processing to assemble coherent sentences | Aimed to improve the quality and comprehensibility of audio signals in noisy environment |
| Training Approach | Trained end-to-end, allowing the model to optimize all components together for direct translation of video to text | Utilizes staged training, first for feature extraction via CNNs, then temporal processing with Bi-LSTMs | Typically involves training on a mixture of clean and noisy audio to learn filtering and noise reduction techniques |
| Accuracy & Performance | Achieved 95.2% accuracy on sentence-level lipreading on the GRID rubric | Best-performing model reached an accuracy of 88.55% for sentence identification and 84.5% for word recognition | In moderate noise conditions, the improvements might be in the range of 5 to 10 dB typically.<br>In challenging environments, the improvement is typically around 3 to 8 dB |

# III. Methodology

In our exploration of automated lip reading research papers, we previously examined models that aim to improve speech intelligence in noisy settings. While these studies provided significant insights into the application of automated lip reading technologies, we have decided to reference and utilise LipNet model as shown in the research paper "LipNet: End-to-End Sentence-level Lipreading" by Yannis M. Assael et al., because of its creative approach and outstanding performance metrics.

Instead of just identifying words or phrases, LipNet sets itself apart from the other research models as the first end-to-end model to be able to predict sentence-level sequences. The Turkish lip-reading model utilises CNN with Bi-LSTM networks and the LipNet model incorporated and combined connectionist temporal features (CTC), recurrent neural networks (RNNs) ,spatiotemporal convolutional neural networks (STCNNs) and Bidirectional Gated Recurrent Unit (Bi-GRU) to successfully integrate spatial characteristics which the Turkish model is unable to achieve. The LipNet model also has empirical data that shows that it outperformed both human lip readings and previously established world-level benchmarks by reaching 95.2% accuracy in sentence-level word recognition on the GRID rubric. After learning that LipNet can perform better than other models that mostly just rely on temporal modeling, we believe that this is the ideal model that perfectly aligns with the objectives of our study.

# IV. Experimental Approach

In our initial implementation of the model, we realised that the training process required a high computational cost resulting in long training times. Even after accelerating the training using a GPU, training the model still took about 12 hours to complete with the resources we had.

This motivated us to simplify our lipreading model to reduce computational costs and training time while preserving its predictive accuracy and reliability. The initial LipNet framework, while powerful, was computationally intensive, incorporating features such as STCNNs and Bi-GRU, but we wanted to streamline the architecture by exploring alternative components that could deliver similar performance with less complexity.

Our first experimental modification involved substituting Bi-GRU with Bi-LSTM networks. While Bi-GRU is effective in managing sequential data, Bi-LSTM provides a more robust mechanism for capturing long-range dependencies due to its enhanced memory cells and gating mechanisms. By implementing Bi-LSTM, we aimed to maintain the model’s ability to integrate temporal characteristics effectively while reducing potential bottlenecks associated with Bi-GRU's training and generalization capabilities.

Our second experimental modification involved simplifying spatial feature extraction. We replaced STCNNs with 2D convolutional layers. While STCNNs excel at integrating spatial and temporal features, they are computationally demanding. Using 2D convolution greatly reduced the complexity of the convolutional network layer, and delegates temporal feature extraction to the recurrent layers. This adjustment provided a more efficient alternative, preserving the model’s spatial understanding while achieving significant computational savings.

Lastly, our third experimental modification replaced the ReLU activation function with Softmax for the classification process. Softmax offers a probabilistic approach to output predictions, enabling a clearer interpretation of the likelihood of each class. This adjustment aimed to enhance the model's prediction reliability, particularly for scenarios requiring distinct classification boundaries. By prioritizing a more interpretable output, we improved the model's usability in practical applications without introducing additional computational burdens.

# V. Experiment & Results

All models were trained to 100 epochs with a learning rate scheduler. For the first 30 epochs, the learning rate remains unchanged. This allows the model to explore the loss surface more effectively during the early stages of training. A stable learning rate in the beginning helps the model converge to a reasonable region in the parameter space. After 30 epochs, the learning rate decreases exponentially with each epoch using the formula:

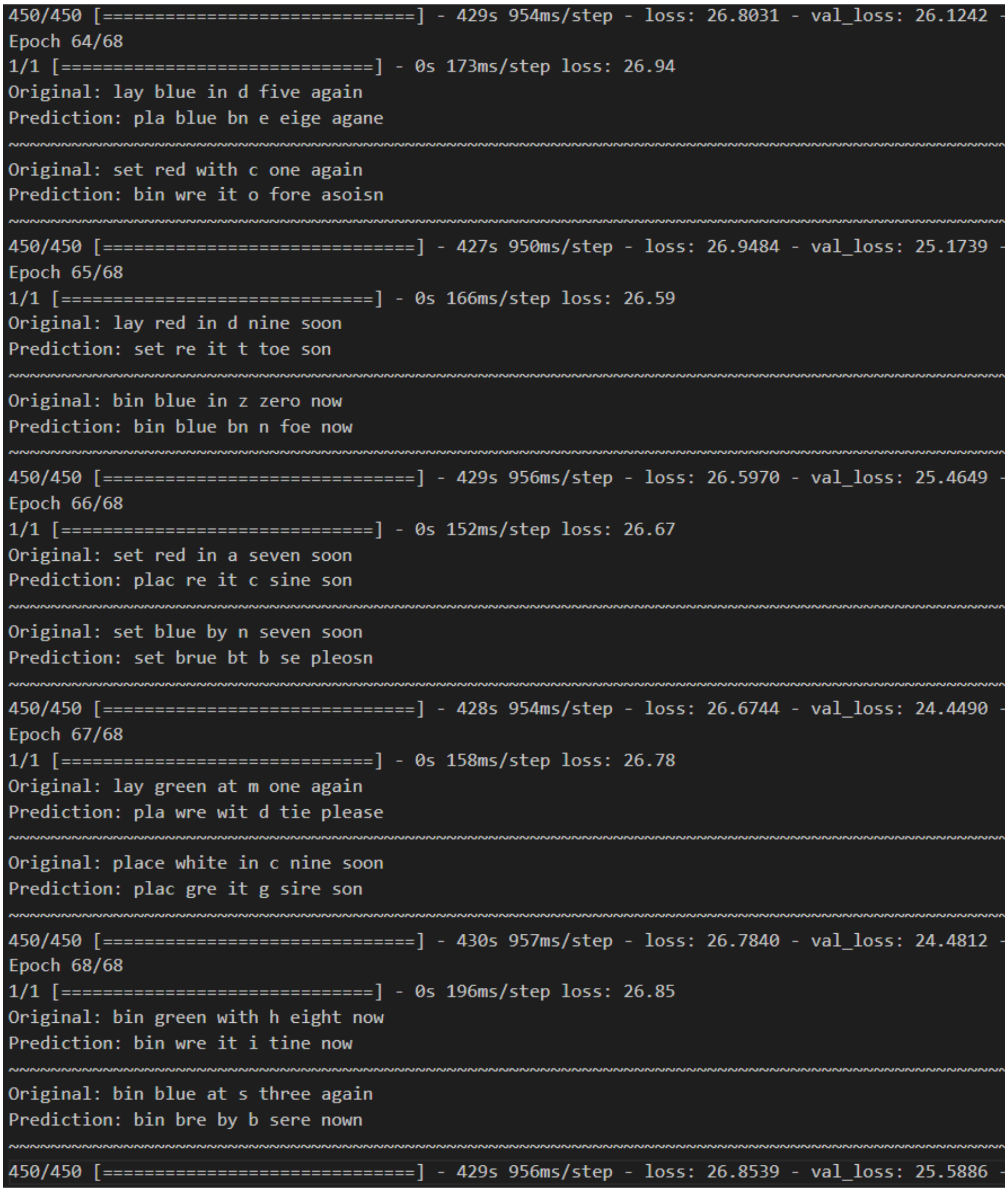
$$new\,LR\ =\ current\,LR\ *\ e^{-0.1}$$

This gradual reduction allows the model to make finer updates to the weights, improving convergence and stability as the model approaches the optimal solution.

A Connectionist Temporal Classification (CTC) loss function is used. It calculates loss between a continuous (unsegmented) time series and a target sequence. CTCLoss sums over the probability of possible alignments of input to target, producing a loss value which is differentiable with respect to each input node. The alignment of input to target is assumed to be "many-to-one", which limits the length of the target sequence such that it must be equal or less than the input length.

A Nvidia RTX 3080 graphics processing unit (GPU) was used to accelerate the training of the models.

The team started off with implementing the model outlined in "LipNet: End-to-End Sentence-level Lipreading" by Yannis M. Assael et al., to evaluate its efficiency and analyse for possible improvements. A model with 3D convolutional layers with Bi-GRU amounting to 4,282,185 parameters.

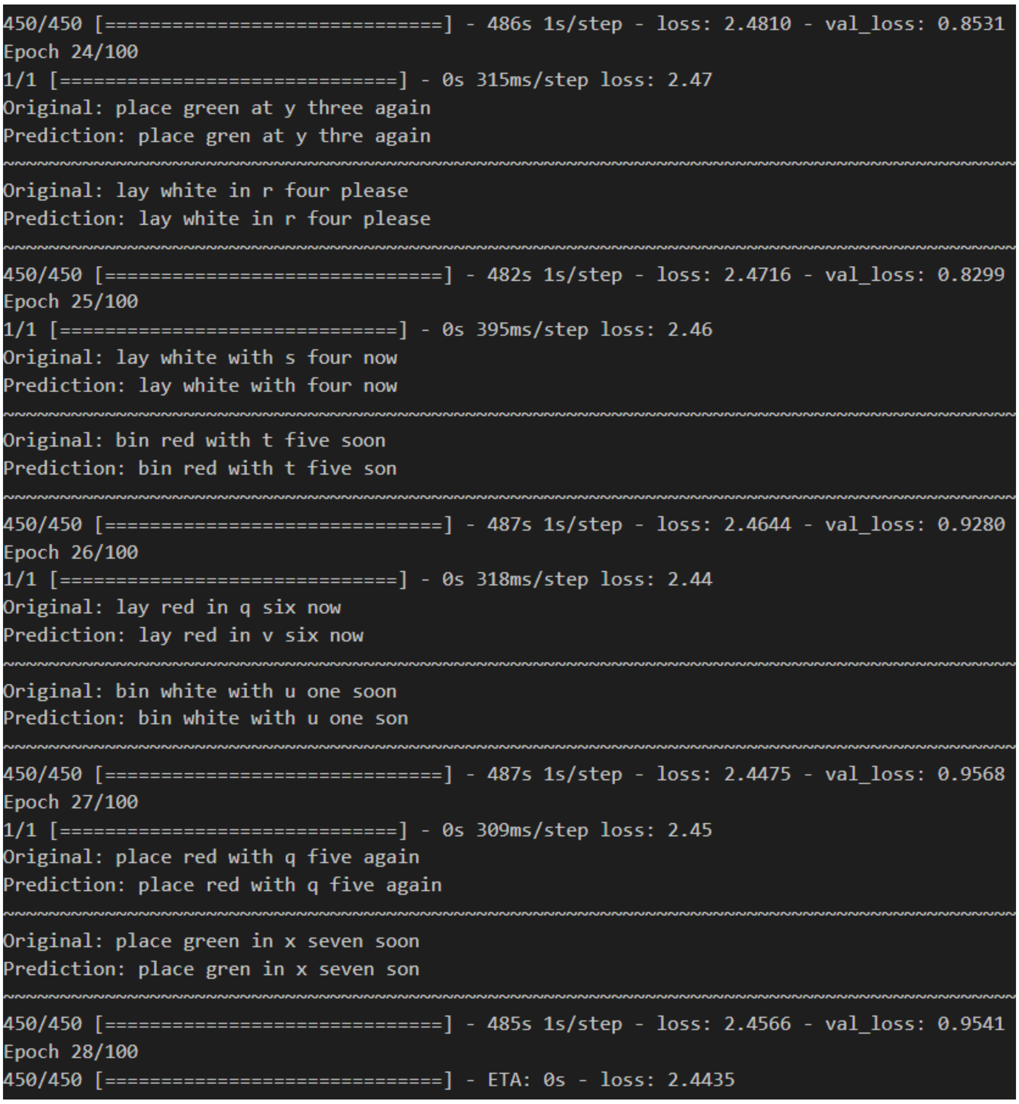


```
450/450 [==============================] - 429s 954ms/step - loss: 26.8031 - val_loss: 26.1242 -
Epoch 64/68
1/1 [==============================] - 0s 173ms/step loss: 26.94
Original: lay blue in d five again
Prediction: pla blue bn e eige agane
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: set red with c one again
Prediction: bin wre it o fore asoisn
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 427s 950ms/step - loss: 26.9484 - val_loss: 25.1739 -
Epoch 65/68
1/1 [==============================] - 0s 166ms/step loss: 26.59
Original: lay red in d nine soon
Prediction: set re it t toe son
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: bin blue in z zero now
Prediction: bin blue bn n foe now
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 429s 956ms/step - loss: 26.5970 - val_loss: 25.4649 -
Epoch 66/68
1/1 [==============================] - 0s 152ms/step loss: 26.67
Original: set red in a seven soon
Prediction: plac re it c sine son
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: set blue by n seven soon
Prediction: set brue bt b se pleosn
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 428s 954ms/step - loss: 26.6744 - val_loss: 24.4490 -
Epoch 67/68
1/1 [==============================] - 0s 158ms/step loss: 26.78
Original: lay green at m one again
Prediction: pla wre wit d tie please
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: place white in c nine soon
Prediction: plac gre it g sire son
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 430s 957ms/step - loss: 26.7840 - val_loss: 24.4812 -
Epoch 68/68
1/1 [==============================] - 0s 196ms/step loss: 26.85
Original: bin green with h eight now
Prediction: bin wre it i tine now
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: bin blue at s three again
Prediction: bin bre by b sere nown
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 429s 956ms/step - loss: 26.8539 - val_loss: 25.5886 -
```

figure 1: LipNet 3D convolutional Bi-GRU training output

As you can see from the figure above, in the last 5 epochs towards the end of training, with 450 iterations per epoch, taking approximately 430 seconds per epoch, an estimated 12 hours was used to train this model. Training loss was stagnant at approximately 26.7 and validation loss at approximately 25.

A test prediction was made after each epoch using a random video sample from the test set to allow analysis of the quality of the model. Predictions took an approximate 170 milliseconds. As seen, the quality of predictions could be improved, which was the aim of the following model that utilizes a Bi-LSTM instead of a Bi-GRU.



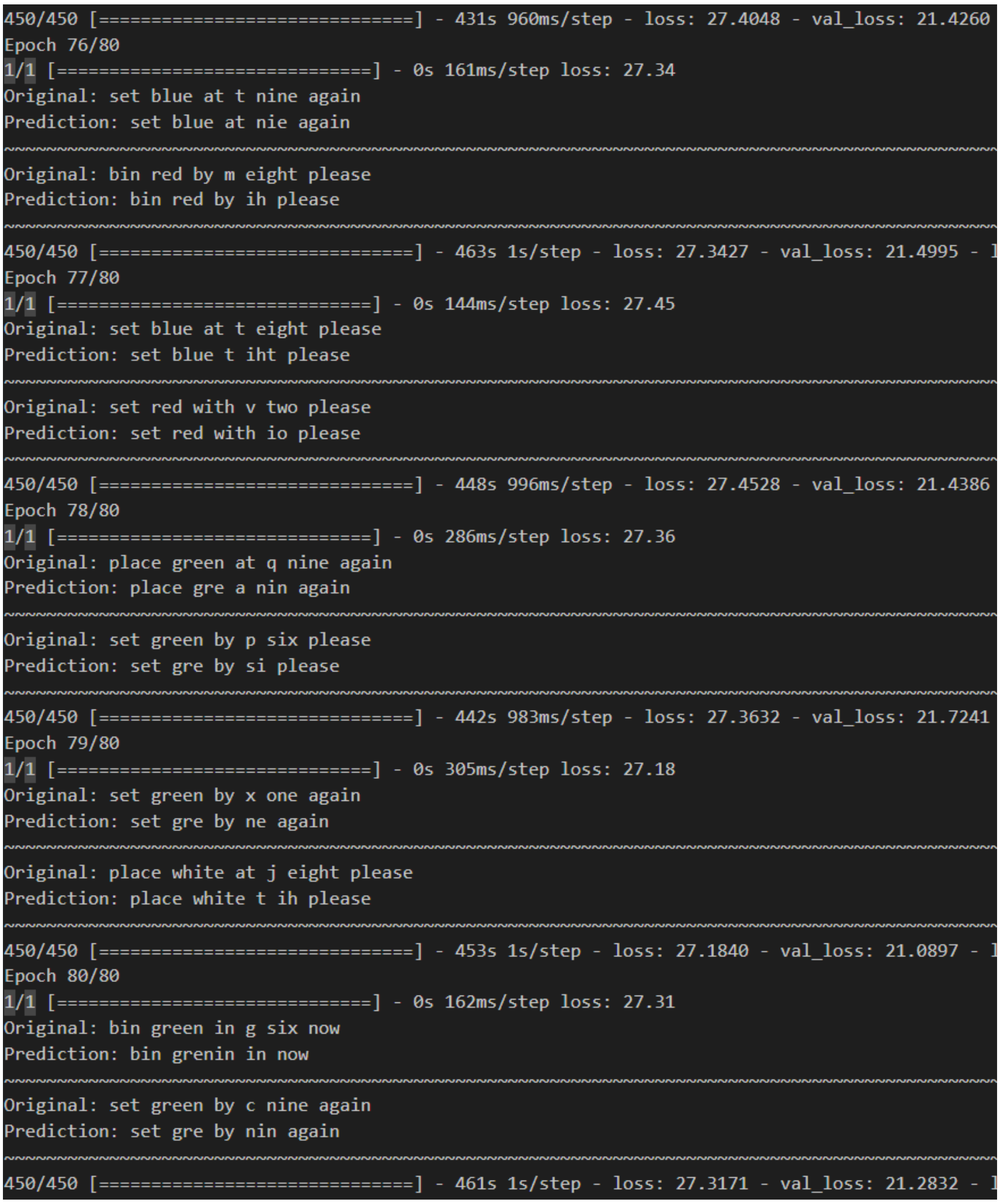
```
450/450 [==============================] - 486s 1s/step - loss: 2.4810 - val_loss: 0.8531
Epoch 24/100
1/1 [==============================] - 0s 315ms/step loss: 2.47
Original: place green at y three again
Prediction: place gren at y thre again
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: lay white in r four please
Prediction: lay white in r four please
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 482s 1s/step - loss: 2.4716 - val_loss: 0.8299
Epoch 25/100
1/1 [==============================] - 0s 395ms/step loss: 2.46
Original: lay white with s four now
Prediction: lay white with four now
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: bin red with t five soon
Prediction: bin red with t five son
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 487s 1s/step - loss: 2.4644 - val_loss: 0.9280
Epoch 26/100
1/1 [==============================] - 0s 318ms/step loss: 2.44
Original: lay red in q six now
Prediction: lay red in v six now
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: bin white with u one soon
Prediction: bin white with u one son
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 487s 1s/step - loss: 2.4475 - val_loss: 0.9568
Epoch 27/100
1/1 [==============================] - 0s 309ms/step loss: 2.45
Original: place red with q five again
Prediction: place red with q five again
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: place green in x seven soon
Prediction: place gren in x seven son
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 485s 1s/step - loss: 2.4566 - val_loss: 0.9541
Epoch 28/100
450/450 [==============================] - ETA: 0s - loss: 2.4435
```

figure 2: LipNet 3D Convolutional Bi-LSTM training

The model includes 8,471,924 parameters and the above figure shows the result of replacing GRU with Bi-LSTM by comparing the last 5 epochs of training. Each epoch took approximately 490 seconds to complete, resulting in an estimated training time of 13.5 hours. The model was significantly improved with training loss and validation loss reduced by a factor of 10 to approximately 2.4 and 0.9 respectively.

The quality of predictions has also increased with relatively little mistakes compared to using Bi-GRU. However, prediction times have almost doubled to an average of 340 milliseconds. This is most likely due to the increase in parameters, with the model more complex and better able to extract features and patterns.

This increase in performance outweighs the relatively small increase in training time and rather large increase in prediction times. With satisfactory accuracy, the team moved to train the next model which utilised 2D convolutional layers instead of 3D layers in hopes of reducing overall training time.



```
450/450 [==============================] - 431s 960ms/step - loss: 27.4048 - val_loss: 21.4260
Epoch 76/80
1/1 [==============================] - 0s 161ms/step loss: 27.34
Original: set blue at t nine again
Prediction: set blue at nie again
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: bin red by m eight please
Prediction: bin red by ih please
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 463s 1s/step - loss: 27.3427 - val_loss: 21.4995 - l
Epoch 77/80
1/1 [==============================] - 0s 144ms/step loss: 27.45
Original: set blue at t eight please
Prediction: set blue t iht please
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: set red with v two please
Prediction: set red with io please
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 448s 996ms/step - loss: 27.4528 - val_loss: 21.4386
Epoch 78/80
1/1 [==============================] - 0s 286ms/step loss: 27.36
Original: place green at q nine again
Prediction: place gre a nin again
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: set green by p six please
Prediction: set gre by si please
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 442s 983ms/step - loss: 27.3632 - val_loss: 21.7241
Epoch 79/80
1/1 [==============================] - 0s 305ms/step loss: 27.18
Original: set green by x one again
Prediction: set gre by ne again
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: place white at j eight please
Prediction: place white t ih please
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 453s 1s/step - loss: 27.1840 - val_loss: 21.0897 - l
Epoch 80/80
1/1 [==============================] - 0s 162ms/step loss: 27.31
Original: bin green in g six now
Prediction: bin grenin in now
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Original: set green by c nine again
Prediction: set gre by nin again
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
450/450 [==============================] - 461s 1s/step - loss: 27.3171 - val_loss: 21.2832 - l
```

figure 3 : LipNet 2D convolutional BiLSTM training

With 5,800,617 parameters, the above figure shows the results of replacing 3D convolutional layers with 2D layers. Despite having a reduced number of parameters compared to the previous model, the training time was relatively unchanged with approximately 450s per epoch, for an estimated total of 12.5 hours. This is similar to the original implementation of the model utilising 3D convolutional layers with Bi-GRU.

The quality of predictions also fell as seen from the increased training and validation losses of approximately 27.3 and 21.2 respectively. The average prediction time is approximately 180 milliseconds.

Although TimeDistributed 2D convolutions are supposedly more computationally efficient as they operate on individual frames independently, have lower memory usage and fewer parameters and are easier to parallelize, the training time was not significantly changed. This is possibly due to hardware limitations as the model was trained on a single GPU and was not able to take advantage of the parallelization capabilities of the model.

# VI. Areas of Improvement

The findings from our experiments highlight several areas where further refinements could enhance both the performance and efficiency of our lip-reading model. These improvements are particularly relevant to our objectives of reducing computational costs and training time while maintaining or enhancing prediction accuracy.

One critical area to explore is the diagnostic analysis of training and validation loss curves, particularly in the Bi-LSTM model. While we observed significant improvements in accuracy with Bi-LSTM over Bi-GRU, the increase in training and prediction time suggests there may be underlying inefficiencies in the current model architecture. By conducting deeper diagnostic analyses, we could identify overfitting or underfitting issues and fine-tune the model's hyperparameters to improve training efficiency.

Another potential area for improvement is weight initialization. The shift from Bi-GRU to Bi-LSTM introduced more complex memory cells and gating mechanisms, which could benefit from a more refined weight initialization strategy to accelerate convergence. Specifically, experimenting with advanced initialization methods like Xavier or He initialization could help the model achieve faster training times while reducing the chances of vanishing or exploding gradients in deeper layers.

In terms of activation functions, the decision to use Softmax over ReLU was made to improve the interpretability of classification results. However, we could explore alternatives such as Leaky ReLU or Swish, which may prevent gradient-related issues, especially with the deeper layers of the Bi-LSTM network. These alternatives could lead to more stable training and improved prediction performance while preserving computational efficiency.

Furthermore, network topology could be optimized by considering the addition of attention mechanisms or residual connections. These approaches have been shown to help the model focus on relevant temporal and spatial features more effectively, potentially enhancing lip-reading accuracy, particularly in noisy or complex video inputs. Adding these mechanisms could offer a balance between performance gains and computational efficiency.

Given that we observed no significant reduction in training time when switching from 3D to 2D convolutions, optimization of the learning process through techniques like adaptive learning rates (e.g., cyclical learning rate schedules or optimizers like AdamW) could be beneficial. These adjustments could reduce the overall training duration and help the model converge more efficiently to a solution, particularly in scenarios involving long sequences of frames.

Finally, regularization techniques such as dropout or batch normalization could help further reduce overfitting, especially in the Bi-LSTM model with its larger number of parameters. These techniques would help improve the generalization ability of the model, ensuring more reliable predictions on unseen data.

By focusing on these areas, we aim to create a more efficient lip-reading model that balances computational complexity with predictive accuracy, aligning with the objectives of our project to optimize training times and enhance performance for real-world applications.

# VII. References

Atila, Ü., & Sabaz, F. (2022, June 29). *Turkish lip-reading using Bi-LSTM and deep learning models*. Engineering Science and Technology, an International Journal. https://www.sciencedirect.com/science/article/pii/S221509862200115X?via%3Dihub

Adeel, A., Gogate, M., Hussain, A., & Whitmer, W. M. (2019). Lip-reading driven deep learning approach for speech enhancement. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *5*(3), 481-490.