Exploration of Visual Speech Recognition with LipNet

Manan Sheth
messagemanan@gmail.com

Abstract

The process of interpreting speech through the observation of lip movements is termed lipreading. LipNet [2] transcribes a variable sequence of video frames into textual representation by employing spatiotemporal convolutions, a recurrent neural network architecture, and connectionist temporal classification loss. When trained on 500 video segments featuring a single speaker and evaluated on 50 distinct video segments from the same speaker using the GRID corpus [4], LipNet achieved a 3% word error rate (WER). This report discusses limitations of GRID data set and further techniques using LLMs to improve lip reading performance. The use of linguistic context from LLMs may really help correcting the predictions from lip reading model output.

1. Introduction

Lipreading is a challenging task for humans, with the accuracy significantly lower for the hearing-impaired people as well. It is particularly hard without contextual cues. LipNet, a deep learning-based automatic speech recognition (ASR) system, addresses this by employing an end-to-end training approach for sentence-level predictions. Operating at the character level, LipNet utilizes spatiotemporal convolutional neural networks (STCNNs), recurrent neural networks (RNNs), and the connectionist temporal classification loss (CTC). We process video frames input and produce a predicted sentence spoken by the speaker as the output.

2. Related work

Since the LipNet paper, there has been research in improving modeling techniques [8], datasets, and also in understanding the nuances of muscle movements. The LRS3-TED [9], [1], [6] papers obtain richer datasets for robust experiments of the machine learning models. [8] discusses a lot more details about setting up large-scale pipeline to produce a sequence of phoneme distributions given a sequence of video frames. It also discusses the challenge of video processing such as computational and memory limits and issues related to stable video frames that are not considered corrupt or impossible to process as tensors due to inherent flakiness.

A significant challenge in lip-reading is that many different sounds are made in the same area of the mouth. This means two different sounds can be captured as nearly identical video frames, which makes them hard to tell apart visually. These types of sounds are known as homorganic sounds. There is another field of research recently being more important and exciting to experiment with is use of LLMs in VSR. Research ideas such as VALLR [9] propose techniques to use linguistic context from LLM to correct the prediction of the LipNet-like model output. The LipNet model completely relies on understanding the video frame and the pixel values to predict spoken character which may miss the contextual awareness about the language, scene setup where the speaker is, and timeline what has happened in the past in video. Using LLM as a tool to refine the output with prompts to help correct the prediction could potentially improve the accuracy drastically in real world scenarios like providing real-time closed captions in the Augmented Reality (AR) glasses for noisy environments at a sport event or surveillance from a magnifying lens.

3. Methods

LipNet implementation begins with three blocks of 3D convolutions. Each block uses a ReLU activation and is followed by a MaxPool3D layer that down-samples the spatial dimensions (height and width) while preserving the full temporal (frame) sequence. After the convolutional blocks, a reshape layer flattens the spatial and channel dimensions for each of the 75 time steps. This prepares the volumetric features for processing by the recurrent layers. The core of the temporal analysis is handled by two stacked bidirectional LSTM layers (similar to the use of Bi-GRU in the original implementation in [2]). Using dropout after each layer helps prevent overfitting. This structure allows the model to learn contextual information from the full sequence of lip movements, looking at both past and future frames. The final stage consists of a dense layer with softmax activation. This layer produces a probability distribution over all possible characters in the vocabulary (plus the CTC blank token) for each of the 75 time steps. The

total parameter count of the entire model is around 8.4M, where the majority of parameters (6.6M) are from the first Bi-LSTM unit.

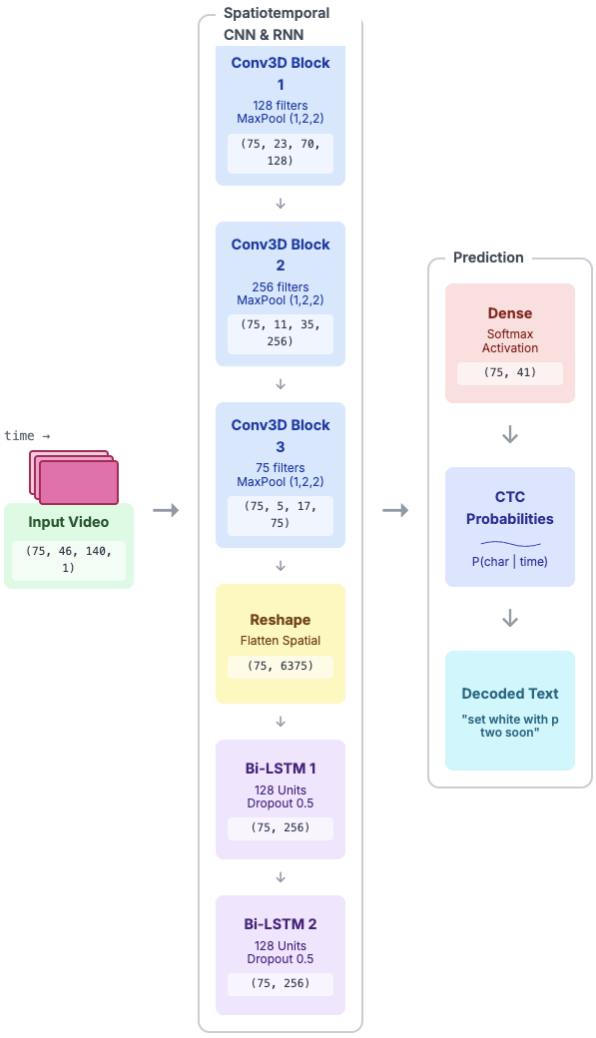

Figure 1. The high-level workflow of implemented system, starting with video input, processing through the LipNet model, and using an CTC loss at the end.

Lip reading is an inherently dynamic process that requires the interpretation of motion. So, LipNet architecture is using Spatio-temporal Convolutions (3D convolutions). 2D convolutions operate only over the height and width of an image, whereas STCNN utilizes 3D kernels that convolve across two spatial dimensions (height, width) and one temporal dimension (the sequence of video frames). This is very useful since it allows the network to learn not only the spatial features of the mouth region in a given frame but also the temporal dynamics of how these features evolve across frames. By learning these motion patterns directly, the STCNN can extract powerful feature representations

Layer (type)	Output Shape	Param #
conv3d (Conv3D)	(None, 75, 46, 140, 128)	3,584
activation (Activation)	(None, 75, 46, 140, 128)	0
max_pooling3d (MaxPooling3D)	(None, 75, 23, 70, 128)	0
conv3d_1 (Conv3D)	(None, 75, 23, 70, 256)	884,992
activation_1 (Activation)	(None, 75, 23, 70, 256)	0
max_pooling3d_1 (MaxPooling3D)	(None, 75, 11, 35, 256)	0
conv3d_2 (Conv3D)	(None, 75, 11, 35, 75)	518,475
activation_2 (Activation)	(None, 75, 11, 35, 75)	0
max_pooling3d_2 (MaxPooling3D)	(None, 75, 5, 17, 75)	0
reshape (Reshape)	(None, 75, 6375)	0
bidirectional (Bidirectional)	(None, 75, 256)	6,660,096
dropout (Dropout)	(None, 75, 256)	0
bidirectional_1 (Bidirectional)	(None, 75, 256)	394,240
dropout_1 (Dropout)	(None, 75, 256)	0
dense (Dense)	(None, 75, 41)	10,537

Table 1. Model summary of implemented LipNet baseline

corresponding to the articulation of phonemes.

The output of the STCNN front-end is a sequence of feature vectors, where each vector represents a slice of time in the input video. To model the dependencies within this sequence, Bidirectional Recurrent Neural Network is used. Standard unidirectional RNN processes the sequence chronologically so its prediction at a given time step is informed only by past context. This is a significant limitation for speech and lip reading, where the identity of a phoneme is often dependent on both past and future contexts (similar to language modeling problems). A bidirectional architecture overcomes this by processing the input sequence in two directions simultaneously: one forward pass from beginning to end, and one backward pass from end to beginning.

To avoid the alignment issue between the video length and the sentence, the Connectionist Temporal Classification (CTC) loss is useful. CTC works by augmenting the set of possible character labels with a special ”blank” token. It removes the need for pre-aligned data and it inherently handles variable-length sequences.

The next iteration was to use the idea from the LipNet paper and obtain a result from a richer and more recent

dataset. For that, WildVSR [6] dataset was selected as it showed a more diverse representation of clip video that is included. The videos have person speaking sentences that we hear in day-to-day life. Since the sentences and frame length are longer in time dimension compared to the GRID dataset videos, the exact architecture that was implemented for the GRID dataset was not possible to run on the WildVSR dataset. For batch size of 1 (single video), out-of-memory (OOM) errors were coming out of the training runs when the T4 GPU memory size was 15GB and each video tensor was requiring around 18GB RAM at minimum.

4. Dataset

LipNet primarily uses the GRID dataset which has video clips with speaker speaking sentences generated from a simple grammar with six word categories: command (bin, lay, place, set), color (blue, green, red, white), preposition (at, by, in, with), letter (A-Z excluding W), digit (zero-nine), and adverb (again, now, please, soon). There are 4 choices for command, color, preposition, and adverb, 25 choices for letter, and 10 choices for digit, resulting in 4 * 4 * 4 * 25 * 10 * 4 = 64,000 possible sentences. Examples of generated sentences include "set blue by A four please" and "place red at C zero again". Each video clip is 75 frames.

To process the video data for our model, code implements a dedicated loading function that reads each video file using OpenCV. For every frame in the video, it first converts it from its original RGB color space to grayscale, reducing the dimensionality of the input. Then each frame was cropped to a specific region of interest, isolating the speaker's mouth area to ensure that the model focuses on relevant visual cues. After processing all frames, thecode performs temporal normalization throughout the video sequence.

Compared to the GRID dataset, the WildVSR dataset is much more aligned to what we hear in daily life in the English language. There is no fixed format for how many words a sentence would have or which type of word would come at which part of the sentence in the WildVSR dataset. In addition to that, the variety of words spoken were very limited in the GRID data set, which made the LipNet model easy to get the accuracy and low WER noted in the paper. The longest length in the WildVSR dataset for a sentence is around 340 English characters, and the approximate average length of the label is around 170 English characters. Similarly, the length of the video (or number of frames) is more than double in the WildVSR dataset compared to the GRID dataset.

In the paper [6] it is mentioned that compared to the LRS3 data set the WildVSR test data set has a higher number of utterances along with 1.5x unique speakers, 4.6x word instances, 3x vocabulary coverage and 5.3x the duration. [6] discusses how popular models such as Whisper [7] and BERT [5] based models perform worse in WER compared to LRS3 [1] indicating difficulty of the data set. Performance degradation is noticed due to the dataset including some non-native speakers, videos with different head poses, and harder vocabulary.

5. Results

Each of the 75 frames of a clip is cropped during the data preparation phase to be of dimension 46 pixel height and 140 pixel width. Training in Google Colab (link) for 70 epochs with around 450 training examples of a single speaker speaking from the GRID data set is able to reproduce the model which during the test of 50 test examples showed the total WER of 1.5, resulting in 3% WER overall. Achieved WER is within the same ballpark of what the LipNet [2] paper mentions in table 2.

The training run of modified LipNet architecture on the WildVSR data set was not finished before the submission date, but for the initial 15 epochs, the training and the validation loss were really high. And there was no evident downward trend noticed in the loss as training progressed from epoch 1 to epoch 15. For WildVSR data set, as mentioned earlier, LipNet architecture was modified slightly in order to fit the training into the 15GB RAM limit of the training instance on Colab. So, there is no fair comparison performed over how LipNet behaves over two datasets.

5.1. Training overfit

While training the model after 30 epoch the training loss was noticed to be declining for the 450 examples but the validation loss started to stagnate at a constant value.

The following graph shows only the loss computation from epoch 32 to showcase the overfit. From epoch 1 to epoch 32 the model loss went dramatically down.

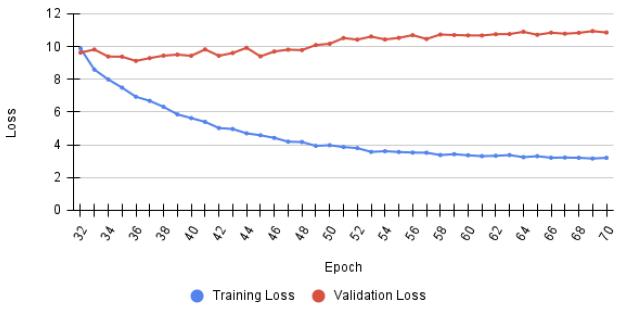

Figure 2. Loss progression after first 31 epochs for both validation and training

6. Future Work

There are a number of possible follow-ups to be tested in the next attempt, such as adding a linguistic context similar

to [9] or, using a transformer based architecture to achieve similar results. And evaluating performance of LipNet and transformers based models on richer datasets like LRS3-TED [1] and [3].

Figure 3. High level block diagram of how a fine tune LLM can be used after a Visual recogition block model output for better prediction accuracy

Above is a simpler diagram to explain the idea from the [9] paper and how it can be generalized and used with LipNet or any other recent architectures like Transformer. The paper explains that a large language model pre-trained for phoneme→sentence reconstruction helps increase the accuracy of the prediction and also eliminates reliance on extensive lip-reading video data for pre-training.

References

[1] T. Afouras, J. S. Chung, and A. Zisserman. Lrs3-ted: a large-scale dataset for visual speech recognition, 2018. 1, 3, 4

[2] Y. M. Assael, B. Shillingford, S. Whiteson, and N. de Freitas. Lipnet: End-to-end sentence-level lipreading, 2016. 1, 3

[3] J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. Lip reading sentences in the wild. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, July 2017. 4

[4] M. Cooke, J. Barker, S. Cunningham, and X. Shao. An audio-visual corpus for speech perception and automatic speech recognition. *The Journal of the Acoustical Society of America*, 120(5):2421–2424, 11 2006. 1

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. 3

[6] Y. A. D. Djilali, S. Narayan, E. L. Bihan, H. Boussaid, E. Almazrouei, and M. Debbah. Do vsr models generalize beyond lrs3?, 2023. 1, 3

[7] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever. Robust speech recognition via large-scale weak supervision, 2022. 3

[8] B. Shillingford, Y. Assael, M. W. Hoffman, T. Paine, C. Hughes, U. Prabhu, H. Liao, H. Sak, K. Rao, L. Bennett, M. Mulville, B. Coppin, B. Laurie, A. Senior, and N. de Freitas. Large-scale visual speech recognition, 2018. 1

[9] M. Thomas, E. Fish, and R. Bowden. Vallr: Visual asr language model for lip reading, 2025. 1, 4