Predicting Emotional OutBERSt: Emotion Recognition From Smartphone Recordings

Danny Nguyen
School of Computing Science
Simon Fraser University
dkn3@sfu.ca
301449063

David Xiong
School of Computing Science
Simon Fraser University
dxa14@sfu.ca
301305081

Tegnoor Gill
School of Computing Science
Simon Fraser University
tegnoorg@sfu.ca
301344408

***Abstract*—People express intense emotions in different ways, and one could shout for many reasons, such as excitement, anger, frustration, surprise, or happiness. How do we determine that someone is screaming from distress and requires assistance in a naturalistic setting? A problem of researching emotion recognition through sound is that the audio is recorded in a static and ideal situation with professional microphone setups. The models trained with that dataset could have biases towards perfect sounds. The BERSt dataset aims to reduce the sound bias by having audio recorded on multiple different devices in various positions. The prompt recordings are conducted by a variety of actors with different language backgrounds. We trained three models of CNN, RNN, and CRNN on MFCCs extracted from the BERSt dataset , which are more robust to noise. We also labeled our perceived emotion on a subset of data to test against the model. While the models perform well on the training data, it has limitations when predicting on the labeled test data. We hope that our findings contributes to the BERSt team and furthers the research of detecting emotions through imperfect sounds.**

I. Introduction

Imagine a person is walking home and they hear someone scream at a nearby park. How would they determine if that scream was coming from someone feeling a positive emotion, or from someone in distress? Humans recognize emotion from a variety of cues, including words, audio, facial expressions, and body language [1]. While there have been many models created in the field of affective computing that can recognize social signals relatively well, accurately pinpointing emotions emotions from speech still requires a lot of research.

The BERSt dataset, or "Basic Emotion Random phrase Shouts" is a dataset created with actors voicing emotional speech using a variety of smartphones in many different positions. BERSt's initial research goal was to study the recognition of shouted and distressed speech from smartphone recordings [2]. A significant challenge of audio emotion recognition today is that most of the recordings created are meticulously curated under controlled conditions. These dataset fail to capture the imperfections, naturalistic, and variability. The BERSt dataset was created with the goal of combating homogeneous speech recorded in a static environment.

There have been research conducted on other sound emotion recognition datasets that are more static in its sound data gathering, such as the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) and the Interactive Emotional Dyadic Motion Capture Database (IEMOCAP). RAVDESS is a multi-modal and dynamic set of facial and vocal expressions. The database was created with a set of gender balanced 24 actors, with the task of speaking prompts in given affects at two different emotional intensities and matching their facial expressions to the emotion [3]. The voice recording was conducted inside a studio, with participants being 20cm away from the microphone in a stationary setting. The IEMOCAP database was recorded by 10 actors with markers on their face, head, and hands to record their facial expressions and body movements when acting out their scripted scenarios, or during spontaneous interactions. Actors would try to elicit authentic emotions to make this database more viable [4].

One research focused on emotion recognition from speech that was conducted on the RAVDESS dataset showed that they had the biggest success in classifying emotions with a Convolutional Neural Network by extracting the Log - Mel Spectrogram (LMS) features with an accuracy of 68% [5]. Their research tested different audio feature extraction techniques to use as features, such as using Mel-Frequency Cepstral Coefficients (MFCC) and LMS. They observed that the choice of audio features impacted the results more than the model complexity. Another research was conducted on both the IEMOCAP and RAVDESS dataset with the goal of improving accuracy and robustness of emotion recognition from speech [6]. This research came to the conclusion that RAVDESS was good as a supplementary dataset as it has poor naturalness.

Our project is an extension to the BERSt research project, and we aim to test the emotion recognition accuracy of multiple different machine learning models on the BERSt dataset. We use extracted MFCCs from the audio as the primary feature and we hope to provide insight to the BERSt team of the consequences of using MFCCs against various machine learning models.

II. Approach

Our feature extraction technique of choice is to use Mel-Frequency Cepstrum Coefficients (MFCC), which is a feature extraction technique commonly used in audio and speech processing as a compact representation of sound [7]. It is derived from the Mel-frequency scale that maps the frequency range

of human hearing into space, which provides a representation that is more aligned with how humans perceive and process sound. For this reason, we use MFCC as the primary training feature to represent our audio data for training the models.

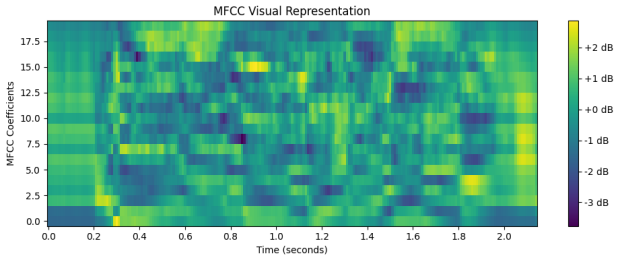

Fig. 1. Visual Representation of MFCC for Neutral Affect Prompt

Another viable option to extract features is to use Mel spectrogram, which is the time-frequency representation of audio data [8]. However, MFCCs are considered more robust to noise, at handling signal framing, and are informative in terms of spectral roll-off. This robustness can therefore lead to a noticeable improvement in our accuracy test. Another reason to why we use MFCC is due to its compact representation, which is more beneficial for our large datasets of over 3000 audio data.

Our approach for using MFCC includes normalizing the signal to ensure that all features has a consistent scale. We also use python's librosa library to load the audio and and extract the features from the normalized signal. On the last extraction step, we use a padding technique in which if the audio file is shorter than the maximum sequence length, we pad it with zeros, and truncate the audio if it is longer.

Following the extraction of MFCC, we have decided to proceed with the three models for the purpose of distinguishing the performance evaluation between different approaches. The first is Convolutional Neural Network (CNN), following by Recurrent Neural Network(RNN), and finally, CRNN - the combination of both models for predicting the affect on our datasets.

CNN model [9]is built using the Keras library and consists of several layers designed to progressively extract higher-level features from the input MFCCs. Each of the used layers in our CNN model can be summarize as the table below:

Layer #	Name	Number of Filters	Kernel Size	Activation	Dropout Rate
1	Conv2D	32	3x3	'relu'	0.25
2	Conv2D	64	3x3	'relu'	0.5
3	Flatten	-	-	-	-
4	Dense	128	'relu'	l2(0.01)	0.5
5	BatchNormalization	-	-	-	-
6	Dense	256	'relu'	l2(0.01)	0.25
7	BatchNormalization	-	-	-	-

Fig. 2. Summary of Layers Used For CNN

Next, we employed the Recurrent Neural Network [10], or specifically a Long Short-Term Memory (LSTM) network. LSTM is a type of RNN that is capable of learning long-term dependencies, making it particularly suited for sequence prediction problems. Since time-series prediction is a difficult task in many fields including sound recognition, traditional methods often struggle with capturing long-term dependencies in the data.

Finally, we implemented the Convolutional Recurrent Neural Network [11], a hybrid model that combines the strengths of CNNs and RNNs, aiming to improve the accuracy of emotion recognition from voice recordings. CRNN is also defined using the Sequential API from TensorFlow's Keras library. In theory, CRNN model is particularly suited to this task because it can handle the temporal dynamics in the audio data (via the RNN component) and also extract local features from the MFCC representations of the audio data (via the CNN component). This combination can allow the model to effectively learn and predict the emotions in the voice recordings.

All three models are first prepared by loading the training and test data from the CSV files containing information about the audio files and their labels. The 'affect' labels are converted to numerical values using a LabelEncoder, and then one-hot encoded. The training data is then split into a training set and a validation set. compiled with Categorical Cross Entropy Loss, Adam optimizer, and Accuracy as the metric. After training, we proceed to use the models to predict the classes of the test data. The accuracy and F1 score of the model are then calculated to evaluate its performance.

III. Dataset

Our project's dataset consists of three main sections: The recorded audio (.wav files), the training data supplied by the BERSt team, and a subset of the training data that we labeled to use as a test set against the models.

TABLE II
ACTOR'S DEMOGRAPHIC INFORMATION

Feature	Values	Count
Age	18-24	19
	25-39	42
	40-60	25
	60+	10
Gender	Female	60
	Male	33
	Non-binary	2
	Undisclosed	1
First language	English	73
	Spanish	6
	Portuguese, French	3
	Chinese(unspecified), Mandarin	3
	Hindi, Croatian, Italian, Russian, Tagalog, Swahili, Hungarian, Norwegian	1
Current language	English	93
	French, Russian, Norwegian	1

Fig. 3. Voice Actor Background [12]

- **Audio Files:** The primary dataset consists of over 3000 audio chunk files provided by the BERSt team. Each chunk is a spoken script that lasts between 1-3 seconds recorded on a wide variety of different smartphones. These scripts are not just random utterances but are carefully designed to invoke and capture a range of emotions. Each audio file has an assigned affect, which is the emotion that was intended to be expressed by the participant during the recording. This can provide a rich and diverse set of data for training our models. Actors were also given instructions on where and how far to place their smartphones to try and emulate different audio recording situations [2].

- **Training Data:** Accompanying the audio files is a CSV file that contains feature labels for each audio chunk that we also have received from the BERSTs team [2]. These labels provide context to the sounds in the audio files and are crucial for training our models. Following is the list of main labels for the csv:
  - Participant's Age Range
  - Gender
  - Phone Model
  - Affect
  - Script
  - Phone Position
  - First Language
- **Labeled Test Data:** From the list of 3000 audio files, a subset of 100 audio files has been randomly selected from the training dataset to be used for testing. This test set introduces a new feature of 'perceived emotion', which is labeled by us using inter-rater agreement. This means that multiple raters independently assess the emotion in each audio file, and their assessments are used to assign the 'perceived emotion' label. The purpose of this step is to provide a more objective measure of the emotion in the audio file, as it is not reliant on the intended affect but on the emotion that is actually perceived by listeners. In addition to 'perceived emotion', we also applied inter-rater to give our perception on the 'valence' and 'arousal' of each sound. The score is to further determine the validity of our inter-rater agreement.

IV. Experiments and Results

- **CNN:** The CNN model was defined with multiple layers including Conv2D, MaxPooling2D, Dropout, Flatten, Dense, and BatchNormalization. The model is compiled with a categorical crossentropy loss function and the Adam optimizer. It was trained with different hyper-parameters, to test the best combination. We ran a for-loop with 2x2 combinations of different batch size and epochs, which resulted in 4 sets of different hyper-parameters listed in the appendix below.

  From all results of different hyper-parameters sets, the results indicate that the model's performance improved over time. Here are some key observations:
  In the initial epochs, the model had a relatively low accuracy on both the training and validation sets. However, as the training progressed, the model's accuracy improved. However, the validation accuracy fluctuated during these epochs, suggest that we are over-fitting our CNN model. If we used a smaller batch size, the result tends to have a lower validation loss but noticeable smaller accuracy.

  Our main goal is to compare each model's results against the 'perceived emotions' we labeled, and the CNN model achieved an accuracy of 32%, with an F1 score of 0.31. We also evaluated the CNN on against the MFCCs of the test data, and figure 4 shows that the CNN achieved good accuracy of 72%. We believe that the large discrepancy in accuracy can be mostly attributed to the perceived emotions being vastly different from the prompted affect.

```
Accuracy: 0.7157360406091371
F1 Score: 0.718704706867617
Confusion Matrix:
[[71  5  0  3  1  1  3]
 [13 49  1  2  2  8  4]
 [ 9  4 65  4  1  5  4]
 [10  1  0 71  0  5  1]
 [17  9  3  3 40  3  2]
 [11  0  4  5  1 56  1]
 [11  0  4  5  1  1 71]]
Classification Report:
              precision    recall  f1-score   support

           0       0.50      0.85      0.63        84
           1       0.72      0.62      0.67        79
           2       0.84      0.71      0.77        92
           3       0.76      0.81      0.78        88
           4       0.87      0.52      0.65        77
           5       0.71      0.72      0.71        78
           6       0.83      0.76      0.79        93

    accuracy                           0.72       591
   macro avg       0.75      0.71      0.72       591
weighted avg       0.75      0.72      0.72       591
```

Fig. 4. Confusion Matrix and Classification Report From CNN Against Test MFCC

- **RNN:** The RNN was evaluated with a 6 layers, including Dropout, Dense, Batch Normalization, and Long Short-Term Memory (LSTM) layers, and compiled with a categorical crossentropy loss function with the Adam optimizer. The LSTM layers is to allow for the RNN model to harness the long-term dependencies by remembering what it outputs previously that's quintessential to an RNN [13]. The RNN model was tested on multiple different epoch and batch sizes. We noticed through experimentation that the validation accuracy of the RNN stabilized between epoch 30 and 40, so the final epoch size selected was 40. The batch size was also experimented on, and was found that a batch size of 50 led to more promising accuracy.
  Similar to the CNN, the increasing validation loss and fluctuating validation accuracy suggests that there was also over-fitting on the RNN model. Finally, the RNN model was tested against the 'perceived emotions' feature that our team labeled together. It yielded an F1 score of 0.345 and an accuracy of 35%, and is further expanded upon in figure 5. The accuracy and F1 scores are slightly higher compared to the CNN, and this is expected due to the sequential structure of the RNN being more beneficial for analyzing sound data.

- **CRNN:** The CRNN is a combination of both the RNN and CNN. Thus the layers used included Convolutional, Dropout, LSTM, Dense, Batch Normalization, Flatten, and Max Pooling. To allow the layers to work together we reshaped the data. Like the previous models, this model was also compiled using categorical crossentropy loss function along with the Adam optimizer. The model is trained on 50 epochs and batch size of 64. Through testing with different epochs the accuracy stabilized around 40-50 epochs and batch size 64. Minor experiments were also carried out with several different

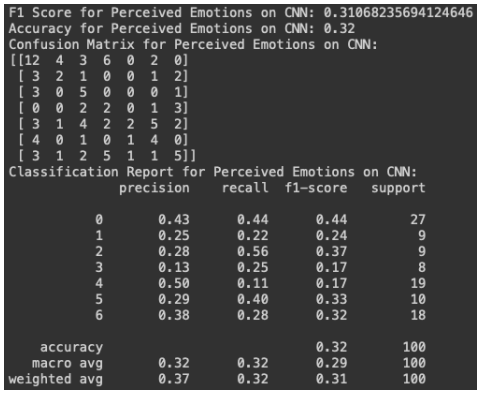

Fig. 5. Confusion Matrix and Classification Report From RNN

	CNN	RNN	CRNN
Accuracy	32%	35%	35%
F1 Score	0.31	0.345	0.36

TABLE I
COMPARISON OF ACCURACY AND F1 SCORES BETWEEN MODELS

designs of CRNN models. The goal of this experiment was to determine which combination of layers would yield in the most optimal result. Key things to note is that reducing flattening resulted in a higher training set accuracy but a lower validation test accuracy. For comparison, reducing the flattening layers increased the training set accuracy from 23% to 92% but also significantly increased the run time.

Testing the CRNN model against the 'perceived emotions' yields an accuracy of 35% and an F1 score of 0.36. Although all relatively close together, the CRNN is our best performing model, beating out the RNN's F1 score by 0.015 while keeping the same accuracy.

V. DISCUSSION

Our models have reported validation accuracy in the range of 14%-25% depending on the epoch. Part of this could just be the margin of error and random outputs, while primarily suggesting over-fitting. When we labeled the 'perceived emotions' using inter-rater agreement, we noticed that our labels only has a 42% match with the given prompt affect. Next, our algorithms achieved a result between 30% - 38% accuracy across all models when tested against our labeled 'perceived emotions'. However, the CNN test against the MFCCs gave a much higher accuracy of 72%. This shows that our model performed poorly against our human perception, and much more accurately against pure MFCC data.

One major method of note in our approach was that we normalized the audio clip lengths. Due to this the length parameter was not a large contributor in determining an emotion but rather the contents of the clip. We achieved this by adding in padding (empty data) at the end of each of our sound chunks to match the length of the longest sound chunk.

While evaluating the performance of the 3 different models (CNN, RNN, CRNN), the hypothesized outcomes were achieved. RNN and CRNN out-preformed CNN. This is most likely due to the dynamic temporal properties of the RNN and CRNN. Both the CRNN and RNN yielded similar results. Dr. Angelica Lim had a different experience of which the CNN out performed the other models. This could be due to the way the data is handled and pre-processed, in combination with padding and truncation. In addition, the experiments mentioned in the introduction section also favoured the results from CNN as opposed to RNN, despite using a different pre-processing technique [5].

Several limitations exist for our project. Our biggest roadblock was the labeling of the 'perceived emotions' and testing the models against it. The process relies on a subjective understanding of the emotion in the short audio chunks. Due to this, bias was introduced, as evident by differences in interpreting accents. Also, the gender, prompt, and first language spoken by the actor affected the perceived emotion and contributes to biases. Next, run time constraints on Google Colab also restricted the exploration of other models that took longer to train than 2.5 hours. Due to this, the possibility of more optimal configurations of the model could exist.
In addition, we were only able to test the models on the extracted MFCCs of the training data. Despite our efforts, we did not have the expertise to train the models on MFCCs as well as other features including 'first language', 'gender', and 'prompt'. For future improvements, we would like to train the model with the aforementioned features alongside the MFCC features, in hopes it would yield a more interesting result.

VI. CONCLUSION

In this project, we have introduced a novel approach to vocal emotion recognition, leveraging a rich dataset and advanced neural network architectures training on extracted MFCCs. Our system trained on over 3000 audio files from the BERSt dataset, demonstrating the capability to discern basic emotions from voice recordings in a naturalistic setting. Despite the challenges posed by the subjective nature of emotion perception, our models achieved interesting results, particularly when utilizing RNN and CRNN architecture. This suggests that the fusion of convolutional and recurrent layers is effective in capturing both spectral and temporal features of emotional speech. The project's significance lies not only in its technical achievements but also in its potential applications, such as enhancing emotional awareness in digital communication. Future work can focus on refining the models and expanding their applicability to a broader range of emotional states and acoustic environments. We hope that this project will be of use to the BERSt team in their future research.

VII. ACKNOWLEDGEMENTS

We would like to thank and acknowledge Paige Tuttosi and Dr. Angelica Lim. Paige is part of the BERSt team and supplied the BERSt dataset as well as gave us the necessary

context for our project. Dr. Lim gave us feedback, advice, and direction on how to approach our project.

VIII. Appendix

A. CNN Hyperparameters

- Batch size 32 and 30 epochs:
  Accuracy: 0.27
  F1 Score: 0.22764981648203025
- Batch size 32 and 50 epochs:
  Accuracy: 0.46
  F1 Score: 0.46442857142857136
- Batch size 64 and 30 epochs:
  Accuracy: 0.32
  F1 Score: 0.28968673805219486
- Batch size 64 and 50 epochs:
  Accuracy: 0.57
  F1 Score: 0.5603118393234672

```
F1 Score for Perceived Emotions on CNN: 0.31068235694124646
Accuracy for Perceived Emotions on CNN: 0.32
Confusion Matrix for Perceived Emotions on CNN:
[[12  4  3  6  0  2  0]
 [ 3  2  1  0  0  1  2]
 [ 3  0  5  0  0  0  1]
 [ 0  0  2  2  0  1  3]
 [ 3  1  4  2  2  5  2]
 [ 4  0  1  0  1  4  0]
 [ 3  1  2  5  1  1  5]]
Classification Report for Perceived Emotions on CNN:
              precision    recall  f1-score   support

           0       0.43      0.44      0.44        27
           1       0.25      0.22      0.24         9
           2       0.28      0.56      0.37         9
           3       0.13      0.25      0.17         8
           4       0.50      0.11      0.17        19
           5       0.29      0.40      0.33        10
           6       0.38      0.28      0.32        18

    accuracy                           0.32       100
   macro avg       0.32      0.32      0.29       100
weighted avg       0.37      0.32      0.31       100
```

Fig. 6. Confusion Matrix and Classification Report From CNN

```
CRNN F1 score on percieved emotions: 0.35658251243080963
CRNN accuracy on percieved emotions: 0.35
CRNN confusion matrix for percieved emotions:
[[9 9 2 2 3 1 1]
 [3 3 1 1 0 1 0]
 [0 0 8 0 0 0 1]
 [0 2 0 1 1 0 4]
 [2 4 4 1 5 3 0]
 [2 3 1 1 0 3 0]
 [1 4 2 2 2 1 6]]
CRNN classification report for percieved emotions:
              precision    recall  f1-score   support

           0       0.53      0.33      0.41        27
           1       0.12      0.33      0.18         9
           2       0.44      0.89      0.59         9
           3       0.12      0.12      0.12         8
           4       0.45      0.26      0.33        19
           5       0.33      0.30      0.32        10
           6       0.50      0.33      0.40        18

    accuracy                           0.35       100
   macro avg       0.36      0.37      0.34       100
weighted avg       0.41      0.35      0.36       100
```

Fig. 7. Confusion Matrix and Classification Report From CRNN

B. Datasheets for Datasets

The answers to the questions of sections 3.1-3.5 in Datasheets for Datasets (https://arxiv.org/pdf/1803.09010.pdf) have been addressed in the paper above.

C. Contributions

- Selecting and Labelling Test Data: All members
- CNN Model: Danny Nguyen
- RNN Model: David Xiong
- CRNN Model: Tegnoor Gill
- Poster Presentation: All members
- Paper: All members

References

[1] T. Pfister and P. Robinson, "Speech emotion classification and public speaking skill assessment," in *Human Behavior Understanding*, A. A. Salah, T. Gevers, N. Sebe, and A. Vinciarelli, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 151–162.

[2] S. BERSt, "Bersting at the screams: Recognition of shouted and distressed speech from smartphone recordings," in *2023 11th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*. Los Alamitos, CA, USA: IEEE Computer Society, sep 2023, pp. 1–5. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ACIIW59127.2023.10388214

[3] S. R. Livingstone and F. A. Russo, "The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english," *PLOS ONE*, vol. 13, no. 5, pp. 1–35, 05 2018. [Online]. Available: https://doi.org/10.1371/journal.pone.0196391

[4] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "IEMOCAP: interactive emotional dyadic motion capture database," *Language Resources and Evaluation*, vol. 42, no. 4, pp. 335–359, Dec. 2008.

[5] K. Venkataramanan and H. R. Rajamohan, "Emotion recognition from speech," 2019.

[6] M. Xu, F. Zhang, and W. Zhang, "Head fusion: Improving the accuracy and robustness of speech emotion recognition on the iemocap and ravdess dataset," *IEEE Access*, vol. 9, pp. 74 539–74 549, 2021.

[7] S. Patnaik, "Speech emotion recognition by using complex MFCC and deep sequential model," *Multimedia Tools and Applications*, vol. 82, no. 8, pp. 11 897–11 922, Mar. 2023.

[8] A. Badshah, J. Ahmad, N. Rahim, and S. Baik, "Speech emotion recognition from spectrograms with deep convolutional neural network," in *2017 International Conference on Platform Technology and Service, PlatCon 2017 - Proceedings*, ser. 2017 International Conference on Platform Technology and Service, PlatCon 2017 - Proceedings. Institute of Electrical and Electronics Engineers Inc., Mar. 2017, publisher Copyright: © 2017 IEEE.; 4th International Conference on Platform Technology and Service, PlatCon 2017 ; Conference date: 13-02-2017 Through 15-02-2017.

[9] B. P. Sowmya and M. C. Supriya, "Convolutional neural network (cnn) fundamental operational survey," in *Intelligent Computing Paradigm and Cutting-edge Technologies*, M. N. Favorskaya, S.-L. Peng, M. Simic, B. Alhadidi, and S. Pal, Eds. Cham: Springer International Publishing, 2021, pp. 245–258.

[10] C. B. M. Karthi and R. Saravanan, "Enhancing the accuracy in classifying human emotion via speech recognition using novel support vector machine compared with recurrent neural network classifier," *AIP Conference Proceedings*, vol. 2822, no. 1, p. 020111, 11 2023. [Online]. Available: https://doi.org/10.1063/5.0172883

[11] E. Cakir, G. Parascandolo, T. Heittola, H. Huttunen, and T. Virtanen, "Convolutional recurrent neural networks for polyphonic sound event detection," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 6, p. 1291–1303, Jun. 2017. [Online]. Available: http://dx.doi.org/10.1109/TASLP.2017.2690575

[12] P. Tuttosi, "Bersting at the screams powerpoint presentation," 2023.

[13] C. Olah, "Understanding lstm networks," Aug 2015. [Online]. Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs/