Dissertation on

"FeelSpeak: Generating Emotional Speech with Deep Learning"

Submitted in partial fulfilment of the requirements for the award of degree of

Bachelor of Technology
in
Computer Science & Engineering

UE20CS390B – Capstone Project Phase 2

Submitted by:

M H SOHAN	**PES1UG20CS235**
RAHUL ROSHAN G	**PES1UG20CS320**
ROHIT ROSHAN	**PES1UG20CS355**
S M SUTHARSAN RAJ	**PES1UG20CS362**

Under the guidance of

Prof. V R BADRI PRASAD
Associate Professor

June December 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
100ft Ring Road, Bengaluru – 560 085, Karnataka, India

PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)
100 Feet Ring Road, Bengaluru – 560 085, Karnataka, India

FACULTY OF ENGINEERING

CERTIFICATE

This is to certify that the dissertation entitled

'FeelSpeak: Generating Emotional Speech with Deep Learning'

is a bonafide work carried out by

M H SOHAN	**PES1UG20CS235**
RAHUL ROSHAN G	**PES1UG20CS320**
ROHIT ROSHAN	**PES1UG20CS355**
S M SUTHARSAN RAJ	**PES1UG20CS362**

In partial fulfilment for the completion of sixth semester Capstone Project Phase 2 (UE20CS390B) in the Program of Study **Bachelor of Technology in Computer Science and Engineering** under rules and regulations of PES University, Bengaluru during the period June. 2023 – December. 2023. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report. The dissertation has been approved as it satisfies the 7^{th} semester academic requirements in respect of project work.

Signature	Signature	Signature
Prof. V R BADRI PRASAD	Dr. Mamatha HR	Dr. B K Keshavan
Associate Professor	Chairperson	Dean of Faculty

External Viva

Name of the Examiners	**Signature with Date**
1. _____	_____
2. _____	_____

DECLARATION

We hereby declare that the Capstone Project Phase 2 entitled **"FeelSpeak: Generating Emotional Speech with Deep Learning"** has been carried out by us under the guidance of Prof. V R Badri Prasad, Associate Professor, and submitted in partial fulfilment of the completion of sixth semester of **Bachelor of Technology** in **Computer Science and Engineering** of **PES University, Bengaluru** during the academic semester January – May 2023. The matter embodied in this report has not been submitted to any other university or institution for the award of any degree.

PES1UG20CS235	**M H SOHAN**	_____
PES1UG20CS320	**RAHUL ROSHAN G**	_____
PES1UG20CS355	**ROHIT ROSHAN**	_____
PES1UG20CS362	**S M SUTHARSAN RAJ**	_____

ACKNOWLEDGEMENT

We would like to express our gratitude to Prof. V R Badri Prasad, Department of Computer Science and Engineering, PES University, for her/his continuous guidance, assistance, and encouragement throughout the development of this UE20CS390B Capstone Project Phase – 2.

We are grateful to the project coordinator, Dr. Priyanka H., all the panel members & the supporting staff for organizing, managing, and helping the entire process.

We take this opportunity to thank Dr. Mamatha HR, Chairperson, Department of Computer Science and Engineering, PES University, for all the knowledge and support we have received from her.

We are grateful to Dr. M. R. Doreswamy, Chancellor, PES University, Prof. Jawahar Doreswamy, Pro Chancellor – PES University, Dr. Suryaprasad J, ViceChancellor, Dr. B.K. Keshavan, Dean of Faculty, PES University for providing us various opportunities and enlightenment during every step of the way.

Finally, this project could not have been completed without the continual support and encouragement we have received from our family and friends.

ABSTRACT

This project presents an innovative approach to imbue synthesized speech with emotion, comprising a two-phase framework: training and testing.

In the training phase, models are developed to detect emotions from labeled text data and to learn emotion-specific pitch, intensity, and modulation from labeled speech data.

The testing phase involves converting input text to neutral speech using text-to-speech (TTS) methods, employing an emotion detection model to discern the emotion in the text, annotating the text with the detected emotion and utilizing a tacotron model to synthesize emotionally expressive speech.

This method exemplifies the integration of machine learning techniques to seamlessly infuse emotion into speech, showcasing the potential for creating emotionally resonant audio from ordinary text inputs. The proposed framework offers a valuable contribution to the field of natural language processing and emotional speech synthesis.

TABLE OF CONTENT

LIST OF TABLES

LIST OF DIAGRAMS

1. INTRODUCTION

In the era of digital communication, where interactions increasingly occur through written text, understanding the emotions conveyed through language has become increasingly crucial. Emotion detection from text, also known as sentiment analysis, has emerged as a powerful tool to decipher and categorize emotions expressed in written form. This burgeoning field holds immense potential to revolutionize human-computer interaction, refine communication strategies, and enhance user experiences.

Emotion detection from text harnesses the power of machine learning and natural language processing (NLP) to analyze text and identify the underlying emotions. By leveraging vast datasets of text labeled with corresponding emotions, these algorithms can effectively classify the emotional content of written language. This capability has opened doors to a myriad of applications across various domains.

Unveiling Emotional Intelligence in the Digital World

The ability to detect emotions from text has far-reaching implications for social media analysis. By analyzing vast troves of social media posts, sentiment analysis can provide valuable insights into public opinion, gauge brand perception, and track the emotional pulse of society. This information can inform marketing strategies, public relations campaigns, and government policies.

Customer reviews offer a goldmine of feedback for businesses, revealing customer satisfaction levels and identifying areas for improvement. Sentiment analysis can automate the process of extracting sentiment from reviews, providing businesses with actionable insights to enhance customer satisfaction and foster loyalty.

Chatbot interactions, increasingly prevalent in customer service and online interactions, can benefit from emotion detection. By understanding the emotional context of user interactions, chatbots can tailor their responses to provide empathetic and personalized support.

Emails, often a primary channel for business communication, can be analyzed to assess customer sentiment and identify potential issues. Sentiment analysis can help businesses prioritize emails, respond promptly to concerns, and maintain positive customer relationships.

Survey responses, often used to gather feedback, can be analyzed to understand the underlying emotions driving responses. Sentiment analysis can provide deeper insights into customer perceptions and inform product development decisions.

Emotion detection from text serves as a crucial step in the process of synthesizing speech infused with emotion. Once emotions are identified in text, they are annotated to the corresponding text. This annotated text is then used to train a speech model, enabling it to learn the prosodic features associated with each emotion. Prosodic features encompass elements such as pitch, intensity, intonation, and rhythm.

By incorporating these emotional cues into synthesized speech, we can create more natural, expressive, and engaging interactions. This capability has the potential to transform various applications, including:

Emotion detection from text has the potential to transform various applications, including educational content, customer service interactions, accessibility tools, entertainment and storytelling, and interactive systems. In education, emotionally charged speech can capture students' attention and make educational materials more engaging. In customer service, emotion-aware chatbots or virtual assistants can provide empathetic and personalized support. For individuals with speech impairments, emotion-aware communication devices can help convey their intended emotions more effectively. Emotionally rich voice-overs can elevate storytelling in movies, games, and other forms of entertainment. Finally, emotion-aware interactive systems can adapt their interactions based on the user's emotional state, providing more personalized and responsive experiences.

2. PROBLEM STATEMENT

This project tackles the issue of conventional text-to-speech systems lacking emotional expression. While these systems can convert text to speech, the resulting output often sounds monotonous or robotic, lacking the emotional nuances needed to convey the intended meaning. This deficiency can lead to confusion and misunderstandings, especially in scenarios where emotions are crucial, like in customer service, counseling, or entertainment.

Our project aims to create a text-to-speech system that infuses emotional cues into the generated speech, aiming to enhance communication and improve user experience. By integrating emotional nuances into the speech, the system seeks to effectively convey the intended meaning of the text, offering practical benefits in fields such as customer service, healthcare, and education.

3. LITERATURE REVIEW

3.1 Literature Review 1

The author Yi Ren [1] tells about the field of Text-to-Speech (TTS) systems has undergone a transformative journey from early concatenative methods to statistical parametric approaches and, more recently, the integration of deep learning and Transformer architectures. The model stands out as a significant advancement in TTS research. Preceding TTS models faced challenges in naturalness, computational efficiency, and controllability. FastSpeech introduces a novel feedforward network based on the Transformer architecture, featuring a sequence-to-sequence (Seq2Seq) model and a length regulator. This innovation enables parallel mel spectrogram generation, addressing the need for real-time synthesis, robust handling of complex sentences, and fine-grained control over synthesized speech characteristics, marking a noteworthy progression in the TTS landscape. FastSpeech presents several notable advantages in the realm of Text-to-Speech (TTS) systems. Firstly, its remarkable speed allows for real-time synthesis, a substantial improvement over traditional methods. The system's robustness shines through its ability to handle lengthy and intricate sentences with finesse, addressing a longstanding challenge in the field. Additionally, the controllability of FastSpeech distinguishes it as a versatile tool, allowing users to adjust not only the speed but also the pitch and volume of the synthesized speech, providing a level of customization unprecedented in TTS systems. Furthermore, its potential challenges in effectively handling languages with complex prosody hint at certain limitations in its universal applicability. Despite these drawbacks, the overall contributions of FastSpeech in terms of speed, robustness, and controllability mark it as a significant advancement in the TTS domain.

3.2 Literature Review 2

The author Rui Liu, [2] introduces a multitask learning (MTL) scheme to address challenges associated with lengthy sentences in Tacotron-based text-to-speech (TTS) systems. The proposed approach extends Tacotron to explicitly model prosodic phrase breaks, focusing on a two-task learning strategy that emphasizes spectral modeling and prosody modeling. By incorporating a dedicated task for phrase break prediction, the authors aim to enhance prosody modeling and improve speech prosody, demonstrating consistent voice quality improvements in both Chinese and Mongolian TTS systems. The effectiveness

of the MTL scheme hinges on the quality of training data, particularly the accuracy of prosody annotations. While the approach showcases enhanced prosody modeling, the inclusion of multitask learning introduces increased model complexity and potential computational resource requirements. The paper introduces MTLTacotron as a novel approach for joint training, providing valuable insights into prosody modeling and highlighting the advantages of multitask learning in Tacotron-based TTS.

3.3 Literature Review 3

The author Papel Chandra at al. [4] proposes a model for detecting emotions in text using big data and deep learning algorithms, specifically the LSTM model. The proposed model involves preprocessing the input data by removing invalid words and extra spaces, and segmenting the data based on various criteria. The paper achieved an accuracy of 0.85 in detecting emotions such as happy, sad, angry, and others. However, it is important to carefully consider the preprocessing step as removing too much information can negatively impact the accuracy of the model.[4]

3.4 Literature Review 4

The author Berrak Sisman [3] speaks out that Tacotron-PL, as presented in this paper, unveils an innovative training strategy for Tacotron-based text-to-speech systems, ushering in a new era in expressive speech synthesis. This groundbreaking approach transcends traditional methods by enhancing speech styling at the utterance level without the need for explicit prosody modeling.

Tacotron-PL leverages the robust Tacotron-based TTS framework to encode the intricate correlation between input text and prosody styles. Unlike conventional approaches, it introduces a unique training paradigm involving frame-level and utterance-level style reconstruction losses. The latter, a perceptual loss carefully formulated to prioritize speech style during training, proves to be a game-changer. Experimental results showcase Tacotron-PL's superior performance, surpassing state-of-the-art baselines in both naturalness and expressiveness.

The author’s methodology behind Tacotron-PL unfolds in three stages, each contributing to its unprecedented success. Initially, a style descriptor is trained using a speech emotion recognizer (SER), capturing deep style features from emotional speech. This descriptor, a product of layers including a 3D

convolutional neural network (CNN), a bidirectional long short-term memory (BLSTM) layer, an attention layer, and a fully connected (FC) layer, becomes the linchpin of style extraction.

Tacotron-PL then undergoes model training with a fusion of frame and style losses, addressing prosody challenges head-on. This comprehensive strategy, encompassing both spectral accuracy at the frame level and perceptual style reconstruction at the utterance level, showcases its potential for expressive speech synthesis without the need for additional modules during runtime.

The style enhancement introduced by Tacotron-PL is propelled by the perceptual loss-based style reconstruction, enabling the model to not just capture but reproduce nuanced prosody styles gleaned from the training data. The unified training strategy, integrating both frame-level spectral loss and utterance-level style loss, marks a significant departure from conventional methods.

4. PROJECT REQUIREMENTS SPECIFICATION

4.1. Objective:

We're embarking on the creation of a system that transforms ordinary text into emotionally charged speech, weaving together the capabilities of the Tacotron model and Text-to-Speech (TTS) methods. This journey unfolds in two pivotal phases: Training and Testing. In the Training Phase, we'll harness the power of the Tacotron model. Armed with a diverse dataset, we'll teach Tacotron the art of generating mel spectrograms from input text, with a keen focus on capturing the emotional nuances. Simultaneously, we're training a machine learning model to detect emotions in text, enriching our system's understanding of the intricate interplay between words and feelings. As we transition to the Testing Phase, the Tacotron model, now finely tuned, steps into action. It transforms input text into mel spectrograms using TTS methods, and concurrently, our trained emotion detection model springs into action, deciphering the emotional context within the text. The ultimate output we seek is a symphony of emotionally expressive speech, where each word resonates with the intended feelings. This synthesis of Tacotron and TTS is more than just a conversion of text; it's an artistic rendering of emotion through speech. So, join us on this journey as we bring words to life, not just audibly but emotionally.

4.2. Training Phase:

4.2.1. Tacotron Model:

We're delving into the intricacies of emotion-infused speech by leveraging a labeled text dataset. This dataset serves as our guide, enriching the Tacotron model during the training process. Tacotron is not just learning to convert text into mel spectrograms; it's diving deep into the emotional landscape. Picture it as adding color to words, where each emotion becomes a distinct hue. But we're not stopping there. We're implementing attention mechanisms, fine-tuning Tacotron to pay extra attention to the emotional nuances embedded in the text. It's like giving the model a sense of focus, ensuring it captures the subtle shifts in tone, emphasis, and rhythm that convey emotions. The result? A Tacotron model that doesn't just reproduce speech

but crafts a symphony of emotion in every mel spectrogram it generates. So, buckle up for a training process that's not just about data; it's about infusing life and feeling into the very fabric of speech.

4.2.2. Emotion Detection Model:

We're upping the ante by training a machine learning maestro to decipher the emotional tapestry within our labeled text dataset. Armed with Support Vector Machines (SVM), Random Forest, and Linear Regression models, our machine learning ensemble is on a mission to become an emotion-savvy virtuoso.Through meticulous training, our system learns to discern and understand the emotional undertones inherent in the provided text. SVM brings its knack for finding complex patterns, Random Forest adds its ensemble wisdom, and Linear Regression contributes its linear insights – together forming a diverse orchestra of emotional interpretation. This training process is more than just algorithms crunching numbers; it's about teaching the system to interpret emotions in a way that aligns with human understanding. So, get ready for a symphony of machine learning, where the notes are emotions and the models are the virtuosos guiding us through the rich landscape of feelings within the text.

4.3. Testing Phase:

4.3.1. TexttoSpeech Conversion:

We're putting the power of TTS (Text-to-Speech) into action by implementing advanced methods. Our secret weapon? The trained Tacotron model. This cutting-edge model is all about turning your input text into mel spectrograms – a detailed representation of the speech that captures its nuances. Imagine it as a finely tuned instrument, taking your words and transforming them into a rich, melodic tapestry. The Tacotron model, having learned from our training phase, is ready to weave your text into a symphony of sound, bringing the emotion and expression to life in the form of mel spectrograms. Get ready to hear your words in a whole new way!

4.3.2. Emotion Detection:

We're unleashing the power of our trained emotion detection model to unravel the feelings embedded in your input text. Armed with sophisticated machine learning techniques like Support Vector Machines (SVM) and Random Forest models, our emotion detection system dives deep into the text, meticulously analyzing it to identify and comprehend the underlying emotions. It's like having a virtual emotional detective, discerning joy, sorrow, excitement, or any other nuanced sentiment. So, with the precision of SVM and the adaptability of Random Forest, our system is geared up to not just read your words but understand the emotions they carry, creating a truly immersive and emotionally resonant experience.

4.4. Output:

We're taking those identified emotions and breathing life into them. Our mission: to generate audio files that don't just carry words but encapsulate the very essence of the desired emotion. Picture it as giving a voice to feelings, turning text into a symphony of speech that resonates with joy, empathy, excitement, or any emotion you desire. The culmination of our Tacotron model, emotion detection finesse, and TTS wizardry results in audio files that aren't just heard but felt. So, get ready to experience your words in a way that goes beyond mere speech – it's a journey into the realm of emotion through sound.

4.5. Usability and User Interface:

We're fine-tuning a user interface that's not just easy but downright delightful to use. Picture this: a space where typing feels as natural as a conversation, seamlessly ushering you into a world of emotionally expressive speech. Consider our interface not merely as a tool but an extension of your own expression, tailor-made for your comfort. Whether you're stationed at your trusted computer, relaxing with a tablet, or on the go with your beloved smartphone, we've got you fully supported. Our aim is to ensure the entire experience is seamlessly smooth and thoroughly enjoyable, regardless of your location or the device you choose. Your ease and satisfaction are at the forefront of our design, making this a personalized and delightful journey, wherever you may be.So, brace yourself to spill your thoughts, and let our user-friendly interface weave them into speech that mirrors the richness of your emotions!

4.6. Performance Metrics:

We're on a mission to nail the art of emotion detection, aiming for precision that captures every subtle nuance with finesse. But our obsession doesn't stop there – we're equally fixated on perfecting the synthesis process, envisioning it as a meticulous tuning of an instrument. Picture it like crafting every note and inflection to be just right, creating a harmonious and expressive result.

And because we understand the value of time, we're not just satisfied with accuracy; we're pushing the boundaries to ensure our system operates in real-time or as close to it as possible. No delays – just an instant and immersive experience that brings emotions to life seamlessly. When it comes to emotions, precision, and speed, we're leaving no stone unturned because excellence is our standard.

4.7. Scalability:

We're forward-thinking, setting the stage for a system that transcends the present and is poised to evolve with the future. Our design philosophy isn't fixed; it's dynamic, ready to embrace growth. Envision a system that effortlessly accommodates new emotions, adapting to an ever-expanding spectrum of feelings that may emerge in the future. We're also preparing to embrace diverse datasets, ensuring our system remains flexible and pertinent as our comprehension of emotions broadens. As emotions evolve and datasets diversify, our system is primed to evolve alongside them, staying at the forefront of advancements in emotional expression.

4.8. Data Requirements:

We're setting the foundation by defining the essential prerequisites for our training datasets. When it comes to labeled text, we're looking for a comprehensive collection that reflects a variety of emotions – a rich tapestry that allows our model to learn the intricacies of each. For speech datasets, clarity and diversity are key. We need a range of voices, accents, and tones to ensure our system can grasp the nuances of emotional expression across different vocal styles. In essence, our minimum requirements are a thoughtful blend of emotion-rich text and diverse speech samples, forming the backbone of effective training for our system.

4.9. Training Model Evaluation:

We're putting the spotlight on accountability by establishing rigorous evaluation metrics to gauge the performance of our Tacotron model, emotion detection algorithms, and regression models. For Tacotron, we're scrutinizing how well it transforms text into mel spectrograms – accuracy and precision are non-negotiable. When it comes to emotion detection, we're measuring our models against their ability to discern emotions accurately from input text. Additionally, for regression models, we're keen on evaluating how effectively they capture the subtle variations in emotional expressions. Our goal is not just accuracy but also assessing how well these models generalize across different scenarios. So, buckle up for a thorough examination that ensures our system stands up to scrutiny and delivers consistent, reliable results.

4.10. Testing and Validation:

We're leaving no stone unturned in ensuring the robustness of our emotional text-to-speech synthesis system. Picture a comprehensive testing plan as a series of meticulously crafted scenarios, each designed to put our system through its paces. We're testing for accuracy, scrutinizing how well our system captures and conveys emotions in diverse text inputs. Effectiveness is on the agenda too – we want to make sure the synthesized speech resonates authentically with the intended emotions. Our plan includes stress-testing under various conditions, assessing real-time performance, and checking for any hiccups in the user interface. It's a thorough examination to guarantee that our system not only meets but exceeds expectations, delivering a seamless and emotionally resonant experience.

4.11. Limitations:

We're being transparent about a few practical aspects of our system. The input text size is capped at 256 characters, ensuring a manageable and focused interaction. While we strive for real-time responses, there's currently a 30-second latency for the output of emotionally nuanced speech. To maintain clarity and precision, our language focus is currently on English US. Additionally, in the realm of emotion detection, we're currently considering a specific,

though limited, set of emotions to ensure accurate and reliable results. These constraints, though noted, are part of our ongoing commitment to refining and enhancing the user experience.

5. SYSTEM DESIGN

5.1 Emotion Detection from Text and Encoding emotions with text

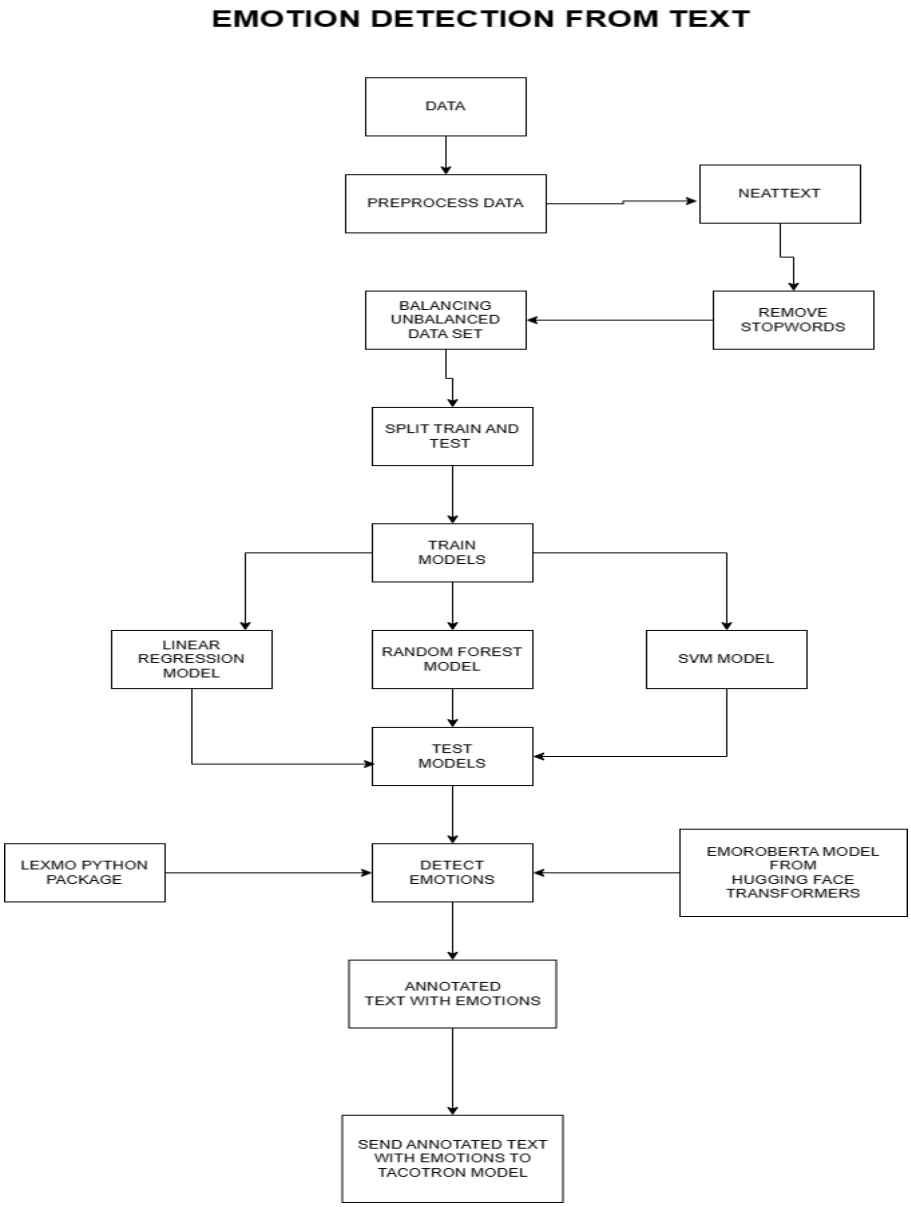

Figure 5.1

Balancing Act: Addressing Dataset Imbalance:

The inherent challenge of an unbalanced dataset beckons our attention next. To ensure the model's robustness across various emotional states, we employ techniques to balance the dataset. This strategic move prevents the dominance of certain emotions, allowing our models to comprehend and discern emotions with equal prowess, irrespective of their prevalence in the dataset.

Data Division: Crafting Training and Testing Grounds:

The dataset, now refined and balanced, undergoes a pivotal division into training and testing sets. This partition ensures that our models are trained on a representative subset while retaining an untouched segment for subsequent evaluation. This duality enables a comprehensive assessment of our models' performance, measuring their ability to generalize beyond the training data.

Model Training: Linear Regression, Random Forest, and SVM:

The training odyssey commences with linear regression, where the model endeavors to discern linear relationships between textual features and emotions. This is followed by the intricate ensemble learning of random forest, capturing nuanced patterns within the text. The geometric precision of the support vector machine (SVM) then takes center stage, carving precise emotional boundaries in the linguistic landscape. Each model undergoes rigorous training, imbibing the intricacies of emotional subtleties present in the dataset.

Model Testing: Unveiling Predictive capability:

The trained models are then put to the test, evaluating their predictive prowess on the testing dataset. This critical phase gauges their ability to accurately classify emotions within unseen textual contexts, providing insights into their real-world applicability and effectiveness.

Advanced Emotion Detection: Lexmo and EmoRoBERTa Integration:

To further elevate our emotion detection capabilities, we integrate advanced tools into our arsenal. Lexmo, a Python package, enriches our analysis with its comprehensive emotion detection capabilities. Additionally, leveraging the power of the EmoRoBERTa model from Hugging Face Transformers, we tap into state-of-the-art natural language processing to enhance the depth and precision of our emotional understanding.

Annotation and Tacotron Integration: Weaving Emotions into Synthesized Speech:

Annotating the text with detected emotions becomes the final flourish in our journey. This annotated text, rich with emotional insights, is then seamlessly integrated into the Tacotron model. The synthesis of speech with embedded emotional nuances marks the culmination of our comprehensive approach to emotion detection from text, transforming linguistic expressions into emotionally resonant synthesized speech.

5.2 Text to Speech with emotion

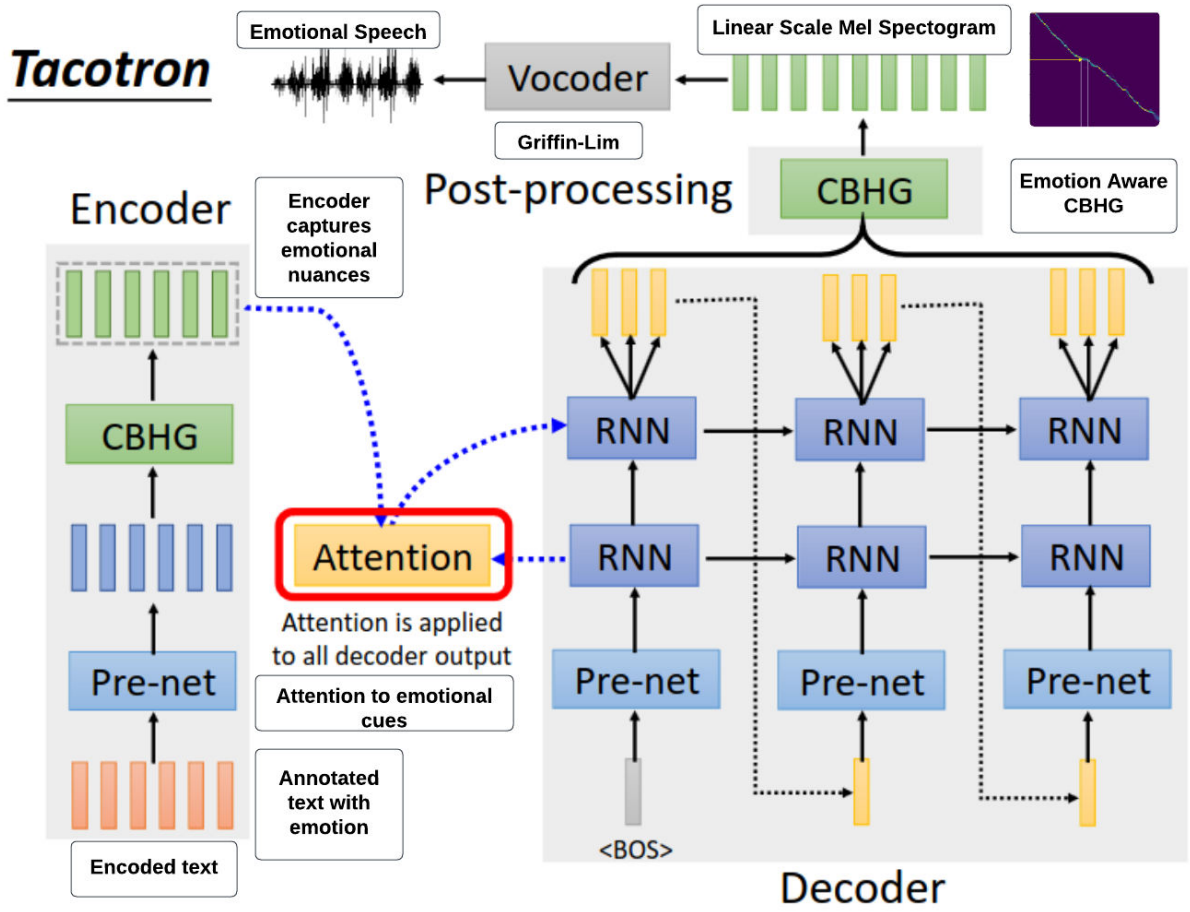

Figure 5.2

Now, let's step into the heart of Tacotron's artistry – the Tacotron Model. Picture it as a seasoned artist, creating not just sound but emotion in every word uttered. This is the one that contributes to the emotional speech synthesis from the text.

Encoder-Emotion-Aware Encoder:

The Encoder, a storyteller with emotional intelligence, delves into the input sequence. Characters and phonemes are not just elements; they are the characters in a novel of emotion. The Encoder, modified to be emotion-aware, captures the subtle nuances, weaving emotion into the fabric of the text. It also utilizes the annotated emotion from the above text model and trains the encoder to understand the emotions for the same.

Attention Mechanism-Emotion-Enhanced:

The Attention Mechanism, a choreographer on the stage, considers not only the text but the emotional undertones. With a dance of alignment, it ensures that emotional cues take center stage, guiding the model in generating a mel spectrogram that resonates with feeling. This connects with the decoder and helps in the model to learn the prosodic features of emotional speech from the speech database and then helps in attaching these learnt prosodic features to the identified text with emotion from the encoder. The results of these are then given to the CHBG module for the further proceedings.

Decoder-Emotion-Conditioned Decoder:

The Decoder, an emotional interpreter, doesn't just decode words; it deciphers emotions. Sensitive to the emotional context, it takes emotion-enhanced encoder representations and attention context vectors, crafting mel spectrogram frames that embody both text and emotion. The hidden states of the Decoder, like a diary, bear the imprints of both textual and emotional information. It has networks of RNNs to implement the prosodic features of the speech to the input text and helps in generating the emotional speech by generating a mel-spectrogram, by defining the waveforms for the corresponding text.

CBHG Module- Emotion-Aware Post-Processing:

Next, the CBHG Module, a maestro refining the final notes. In this emotional post-processing, it doesn't just enhance features; it understands emotions. Convolutional Bank 1D, Highway Network, Bidirectional Gated Recurrent Unit – a trio of virtuosos working together to capture emotional characteristics in the mel spectrogram representation. The CBHG module helps in generating the corresponding mel-spectrogram which is the final spectrogram for the emotional speech.

Vocoder - Emotional Speech Synthesizer:

Finally, our vocoder concludes our model by generating the emotional speech from the resulting mel-spectrogram using the Griffin-lim algorithm. By this, the whole tacotron model comes to the conclusion of generating the emotional speech from text.

6. PROPOSED METHODOLOGY

Proposed Methodology: Integrating Emotion Detection into Text to Speech Synthesis

6.1. Emotion Detection from Text

In advancing emotion detection from text, machine learning algorithms like Support Vector Machine, Random Forest, and Linear Regression play a key role, providing valuable insights into emotion classification. To boost accuracy, integrating state-of-the-art pretrained language models such as EmoRoBERTa has become pivotal. These advanced models, enriched by extensive training on diverse data, enhance the precision of emotion detection by capturing nuanced contextual information within textual expressions. The synergy between traditional machine learning methods and cutting-edge language models signifies a powerful approach, elevating the effectiveness of emotion classification in text.

6.2. Enhanced Data Preprocessing

In enhancing data quality, we employ advanced techniques such as sentiment analysis and contextual language processing for more nuanced cleaning. Additionally, we broaden our dataset by integrating diverse sources, capturing a wider range of emotions and expressions. This dual approach ensures a more comprehensive and representative dataset, laying the groundwork for robust analyses.

6.3. Feature Engineering for Emotional Context

As we delve into the realm of text analysis, our experimentation takes a nuanced turn, focusing on feature representation techniques designed to explicitly capture the emotional nuances embedded in the text. In this endeavor, we go beyond conventional approaches, exploring the integration of sentiment scores, word embeddings, and other features that intricately enhance the representation of emotional context. By incorporating these diverse elements, our aim is to create a more holistic and refined understanding of the emotional fabric within the textual data. This deliberate and thoughtful approach to feature representation underscores our commitment to unveiling deeper insights into the emotional landscape encapsulated in the text under scrutiny.

6.4. Advanced Model Architectures

In our pursuit of advancing emotion detection, we're delving into the realm of deep learning architectures tailored for this task. Specifically, we're exploring the potential of Recurrent Neural Networks (RNNs) and Transformer models, leveraging their capabilities to unravel intricate emotional nuances within the text. To enhance our model's discernment, we're implementing attention mechanisms that dynamically adjust, drawing insights not only from the textual content but also from the identified emotions themselves. This strategic integration promises a more nuanced and contextually aware approach to emotion detection, aligning with our commitment to harnessing cutting-edge technologies for comprehensive and accurate analyses.

6.5. Transfer Learning and FineTuning

We leveraged pretrained emotion detection models and finetune them on the specific dataset to boost performance. We Explored transfer learning techniques to apply knowledge learned from a general emotion dataset to the specific domain.

6.6 Validation and Evaluation Metrics

In our effort to assess the performance of our emotion detection model, we are expanding our evaluation metrics to include precision, recall, and F1 Score, providing a more nuanced understanding of its capabilities. Beyond the conventional measure of accuracy, these metrics enable us to delve into the model's ability to accurately identify and distinguish emotions. Concurrently, we recognize the importance of real-world adaptability and, as such, have introduced a validation set to ensure the robustness of our model across diverse scenarios. This dual approach, encompassing both nuanced evaluation metrics and real-world validation, underscores our commitment to delivering an emotion detection system that excels not only in correctness but also in practical applicability and reliability..

6.7 Emotion Annotation in Text

Expanding our emotion annotation process, we now encompass a broader spectrum of emotional expressions within textual content. This extension enables a more comprehensive understanding of nuanced emotions, ensuring our model can adeptly navigate complex emotional scenarios and

subtle shifts. Implementing a robust mechanism facilitates the recognition of intricate emotional nuances, enhancing the model's sensitivity to the dynamic and intricate emotional landscape present in the diverse range of texts it encounters.

6.8. Tacotron Model Enhancement

In our attempt to enhance the Tacotron model for synthesized speech, we're implementing a significant upgrade. This involves incorporating an emotion-aware encoder, attention mechanism, and decoder, effectively infusing emotional context into the synthesis process. Through meticulous fine-tuning using annotated emotional text, we're ensuring that our model becomes adept at capturing and reproducing nuanced emotions in synthesized speech. This strategic evolution marks a pivotal step forward, promising to imbue our speech synthesis system with the ability to convey not just words but also the rich tapestry of emotions embedded within the text.

6.9 Training on Diverse Datasets

In order to refine the capabilities of our training dataset for speech synthesis, we're taking a comprehensive approach. By expanding the dataset to encompass a diverse array of emotional speech patterns sourced from databases such as IEMOCAP, LJ Speech, and audio books, we're enriching the model's exposure to various emotional contexts. Striving for balance, we ensure that the emotional diversity in the training data is meticulously curated. This meticulous curation guarantees that our model is equipped to handle and replicate a broad spectrum of emotions effectively, enhancing its proficiency in synthesizing emotionally nuanced speech.

6.10 Hyperparameter Optimization

In our attempt to refine model performance, we're delving into the intricacies of hyperparameter tuning. Through a meticulous process, we aim to identify the optimal configurations for both emotion detection and Tacotron model training. This involves fine-tuning key parameters to enhance the models' responsiveness and accuracy. Additionally, we're implementing dynamic learning rate schedules and effective regularization techniques. These measures are designed to

facilitate smoother model convergence, ensuring that our systems learn and adapt optimally, ultimately contributing to heightened precision and proficiency in both emotion detection and Tacotron-based speech synthesis.

6.11 Iterative Testing and Improvement

We have implemented an iterative testing and improvement cycle, gathering feedback from synthesized speech samples and refining the models accordingly. We have also regularly evaluated and updated the models as new data became available.

6.12. User Feedback Integration

We tried to incorporate user feedback and preferences in the training process to tailor the model to user expectations and preferences. We continuously refined the models based on user experience and perception of emotional speech synthesis. By integrating these proposed methodologies, the overall system aims to deliver an advanced and emotionally intelligent text to speech synthesis experience.

7. IMPLEMENTATION AND PSEUDOCODE

7.1 Emotion Detection from Text and Encoding emotions with text

7.1.1 Data Preprocessing

The dataset has 34,793 rows of data with emotion and text. The data preprocessing stage involves cleaning and preparing the text data for analysis. This includes handling missing or invalid data, removing irrelevant or invalid words, and normalizing the text format.

1. Data Cleaning:
    - Remove missing or invalid data entries.
    - Check for and correct typos or spelling errors.
    - Remove unnecessary punctuation, symbols, or special characters.
2. Text Normalization:
    - Convert text to lowercase.
    - Remove extra spaces or whitespaces.
    - Tokenize the text into individual words or phrases.
    - Handle slang, abbreviations, and emojis appropriately.
3. Feature Extraction:
    - Extract relevant features from the text, such as word frequencies, ngrams, or sentiment scores.
    - Represent the text data in a numerical format suitable for machine learning algorithms.

7.1.2 Emotion Detection with SVM

Support Vector Machines (SVM) are a powerful classification algorithm that can be used for emotion detection. SVMs work by finding the hyperplane that best separates the data points into different emotion classes.

1. Model Training:
    - Split the preprocessed data into training and testing sets.

- Train the SVM model using the training set, optimizing the hyperparameters for best performance.

2. Emotion Prediction:
   - Use the trained SVM model to predict the emotion labels for the testing set.
   - Evaluate the model's performance using metrics like accuracy, precision, recall, and F1 Score.

7.1.3 Emotion Detection with Random Forest

Random Forest is an ensemble learning algorithm that combines multiple decision trees to improve classification accuracy. Random Forests are robust to noise and overfitting, making them suitable for emotion detection tasks.

1. Model Training:
   - Split the preprocessed data into training and testing sets.
   - Train a Random Forest model using the training set, optimizing the parameters for best performance.
2. Emotion Prediction:
   - Use the trained Random Forest model to predict the emotion labels for the testing set.
   - Evaluate the model's performance using metrics like accuracy, precision, recall, and F1 Score.

7.1.4 Emotion Detection with Linear Regression

Linear Regression is a statistical method that models the relationship between a dependent variable (emotion score) and one or more independent variables (text features).

1. Model Training:
   - Split the preprocessed data into training and testing sets.
   - Train a Linear Regression model using the training set, optimizing the coefficients for best fit.
2. Emotion Prediction:
   - Use the trained Linear Regression model to predict the emotion scores for the testing set.

7.1.5 Comparative Analysis

Compared to the all three models Random forest model has achieved 89% , SVM and Linear Regression model also achieved 87% and 86% respectively. EmoRoberta model from hugging face transformers have f1-score of 49.30%.

7.1.6 Explanation of code (Emotion Detection From Text)

For detecting emotion from text we have trained and tested many models like linear regression ,SVM, Random forest classifier, EmoRoberta model from Hugging face transformers and LeXmo python packages.

The code uses neat text for text preprocessing and scikit-learn libraries for machine learning tasks. It employs a pipeline approach to streamline the text processing and modeling steps, utilizing CountVectorizer for feature extraction and different classifiers like LogisticRegression, RandomForestClassifier, and SVC for emotion prediction. It also employs RandomOverSampler from imblearn to balance the class distribution and improves the model's performance. The evaluation metrics used to assess the models' effectiveness include confusion matrix, accuracy, precision, recall, and F1-score. Finally, it saves the trained models using joblib for future use.

EmoRoBERTa is a fine-tuned Roberta language model specifically designed for emotion detection from text. Built upon the pre-trained Roberta model, EmoRoBERTa incorporates an additional layer trained on a large dataset of text labeled with various emotions. This allows the model to effectively capture and classify emotional nuances in text, making it a valuable tool for sentiment analysis and emotion-based NLP applications.

LexMo, a Python package specifically designed for text emotion classification, provides a dictionary of emotions and percentages of emotions recognized. This feature offers insights into the emotional distribution within the text, allowing for a more comprehensive understanding of the emotional content. By utilizing both EmoRoBERTa's classification capabilities and LexMo's emotion distribution insights, a more nuanced and accurate analysis of text emotion can be achieved.

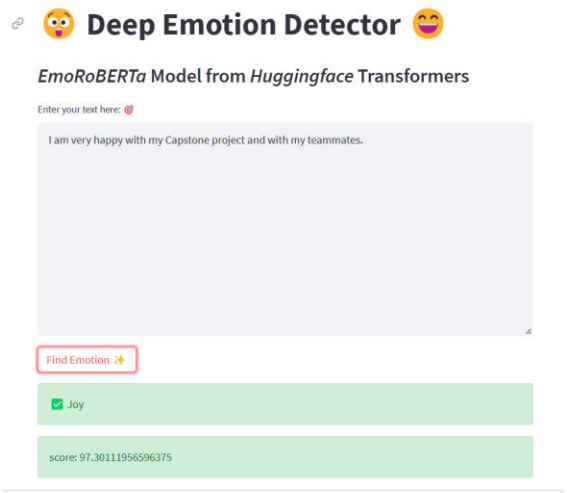

Figure 7.1.1 Output 1

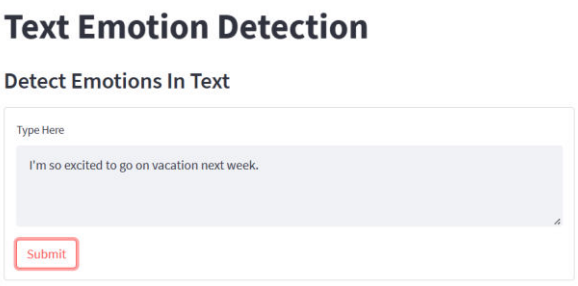

Figure 7.1.2 Output 2

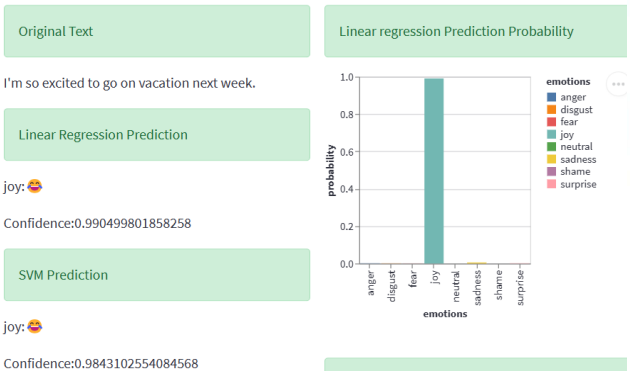

Figure 7.1.3 Output 3

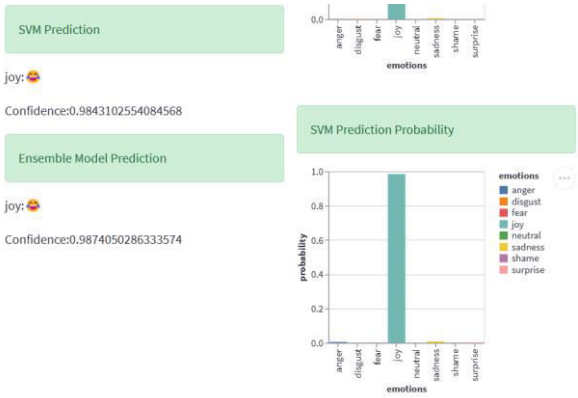

Figure 7.1.4 Output 4

7.2 Annotating emotions with text

- As the emotion is detected from the text, the emotion should be attached to the text part as annotated so that when this annotated file is passed to the tactron model it helps to generate emotional speech. Below figure shows how it is saved in the file.

1	hi i am enjoying my summer holidays!!!!! <joy>
2	
3	i dont like him because of his behaviour so i am dont talk to him <disgust>
4	
5	hello there can i help you with somthing. <sadness>
6	
7	hello there can i help you with something i am bored. <sadness>
8	
9	I am feeling grateful and thankfu <joy>

Figure 7.2

7.3 Text to Speech with emotion

7.3.1 LJ Speech, IEMOCAP and Audio Books Dataset Overview

1. Purpose and Usage:

The **LJ Speech dataset** is preprocessed using a Python script to generate mel and linear spectrograms. The dataset includes audiobooks, with a specific subset processed by the code. We have also used the **IEMOCAP** database. It is annotated by multiple annotators into categorical labels, such as anger, happiness, sadness, neutrality, as well as dimensional labels such as valence, activation and dominance. Also, it contains approximately 12 hours of audiovisual data, including video, speech, motion capture of face, text transcriptions.We have also selected audiobooks as a training database, which are

"A Tramp Abroad"

"The Man That Corrupted Hadleyburg"

Together, these 3 varieties of datasets are used for further training of the model.

2. Dataset Characteristics:

- Audio Data: Consists of approximately 13,000 (for LJ Speech) short audio clips and IEMOCAP has 12 hours of audiovisual data with emotions representing happiness, sadness and anger.
- Format: WAV format with a sample rate of 22.05 kHz.
- Speaker: All recordings are from a single female speaker for LJ Speech and multispeaker for IEMOCAP
- Text Transcriptions: Each audio clip is paired with a corresponding text transcript.

3. Content and Source:

Text Source: Passages from 2 nonfiction books were chosen as the content for the dataset.

Speaker Identity: The dataset features a female speaker with clear and neutral English pronunciation.

4. Example Entry:

- Audio File: lj0010001.wav
- Transcription: "Printing, in the only sense with which we are at present concerned, differs from most if not all the arts and crafts represented in the Exhibition..."

7.3.2 Dataset Preprocessing

1. Datasets and Audiobook Selection:

- Datasets such as LJ Speech and IEMOCAP are used.
- Audiobooks such as "A Tramp Abroad" "The Man That Corrupted Hadleyburg" are used.

2. Parallel Processing:

- The `build_from_path` function utilizes a `ProcessPoolExecutor` to parallelize the preprocessing across multiple worker processes.

3. Metadata Parsing:

- Metadata for the selected audiobooks is loaded from the `metadata.csv` file.
- The metadata includes information about each audio file, such as the filename, duration, and associated text.

4. Spectrogram Generation:

- The `_process_utterance` function is responsible for loading the audio, trimming silence, and generating both linear and mel spectrograms.
- Spectrograms are saved to the specified output directory.

5. Data Feeding:

- The `DataFeeder` class feeds batches of data into a TensorFlow queue on a background thread.
- The data includes inputs, input lengths, mel targets, and linear targets.

7.3.3 HyperParameters

- Hyperparameters for a speech synthesis model are defined using TensorFlow. Speech synthesis involves converting text into spoken language, and the model's hyperparameters are essential settings that influence its architecture, training, and behavior.

1. Text Cleaning:

- The `cleaners` parameter specifies the text cleaning methods applied before training and evaluation.

2. Audio Parameters:

- Describes the characteristics of the audio used in the training data, such as the number of mel spectrogram bins, sample rate, frame length, and pre-emphasis.

3. Model Architecture:

- Defines parameters related to the architecture of the speech synthesis model, including the depth of embedding layers, prenet layers, encoder, postnet, attention mechanism, and decoder.

4. Training Parameters:

- Specifies settings for the training process, such as batch size, Adam optimizer parameters (beta1 and beta2), initial learning rate, and whether to decay the learning rate during training.

5. Evaluation Parameters:

- Sets parameters for the evaluation phase, including the maximum number of iterations, the number of iterations for the GriffinLim algorithm (used in mel spectrogram inversion), and a power parameter.

6. Debugging Function:

- The `hparams_debug_string()` function generates a formatted string containing all hyperparameter values. This function is useful for debugging and reporting, providing an overview of the configured settings.

7.3.4 Models

- Implements the Tacotron model for speech synthesis.
- Components include embeddings, encoder, attention, decoder, post processing CBHG, loss calculation, and optimizer.
- The model is initialized for training or inference based on mel and linear targets.
- Functions for adding loss and optimizer operations are included.

7.3.5 Utilities

1. Audio Processing Utilities :

- An audio processing functions, including loading and saving WAV files, pre-emphasis, and spectrogram generation.
- It supports GriffinLim algorithms for spectrogram waveform inversion both in librosa and TensorFlow implementations.
- There are methods for linear to mel spectrogram conversion and finding the endpoint of a waveform based on silence detection.

2. Signal Processing and Spectrogram Functions :

- The script includes functions for ShortTime Fourier Transform (STFT) and Inverse STFT in both librosa and TensorFlow implementations.
- Mel spectrogram, linear spectrogram, and their inverses are computed using these functions.
- Methods for converting between linear and mel representations, amplitude to decibel conversion, and normalization/denormalization are present.

3. Logging and Plotting Utilities :

3.1 Logging

- A logging system to track and document the progress of training runs. It provides functions to initialize logging, write log messages.

3.2 Utility

- A utility for visualizing alignment matrices as plots.

3.3 Plotting Function

- The `plot_alignment` function generates a plot from an alignment matrix, providing a visual representation of the alignment between input and output sequences during training.

7.3.6 Synthesizer

1. Synthesizer Class:

- Loads a Tacotron model checkpoint and initializes a TensorFlow session.
- Provides a `synthesis` method to generate speech waveform from input text.

2. Model Loading and Initialization:

- Constructs Tacotron model graph with placeholders for text input and lengths.
- Restores model weights from a specified checkpoint during initialization.

3. Text to Speech Synthesis:

- Converts input text to a sequence of phonemes using specified cleaner functions.
- Feeds the sequence to the Tacotron model and obtains a linear spectrogram.
- Converts the spectrogram to waveform, applies pre-emphasis, and trims silence.
- Returns the synthesized waveform as a byte stream.

4. TensorFlow Session:

- Creates and initializes a TensorFlow session for model inference.

5. Audio Processing Utilities:

- Utilizes audio processing functions like waveform inversion, pre-emphasis, and endpoint determination.

- Saves the synthesized waveform as a byte stream (WAV file in memory) for further use.

7.3.7 Training

1. Training Script:

- Performs training of the Tacotron model on a specified dataset.
- Manages data loading, model creation, optimization, and checkpointing.

2. Command Line Arguments:

- Accepts various arguments such as dataset path, model choice, and hyperparameter overrides.
- Allows restoring training from a specific step and setting intervals for summaries and checkpoints.

3. Git Commit Verification:

- Optionally checks if the local Git client is clean and logs the commit hash for version tracking.

4. Data Loading and Model Initialization:

- Sets up a data feeder to load training data.
- Creates a Tacotron model, initializes it with the feeder's inputs, and adds loss and optimization operations.

5. TensorFlow Session and Saver:

- Initializes TensorFlow session and Saver for saving and restoring model checkpoints.

6. Training Loop:

- Iterates over training steps, optimizing the model and logging training statistics.
- Saves model checkpoints at specified intervals and writes summaries for monitoring.

7. Exception Handling:

- Catches exceptions during training, logs the error, and requests the coordinator to stop.

8. Logging and Visualization:

- Logs training information to a log file.
- Writes summaries for TensorBoard visualization.
- Saves audio samples and alignment plots for inspection during training.

9. Main Function:

- Parses command line arguments.
- Initializes logging and hyperparameters.
- Calls the `train` function with specified parameters.

8. RESULTS AND DISCUSSION

8.1 For the text model

In conclusion, this research demonstrated the effectiveness of machine learning algorithms in classifying emotions expressed in text. Random forest model emerged as the most accurate model for emotion detection , but Random forest achieved a superior accuracy score compared to SVM and Linear Regression. The findings highlight the potential of machine learning for understanding human emotions and developing applications that can analyze and respond to emotional cues in text based interactions.

Text	Linear Regression	SVM	Ensemble model	EmoRoberta model (Hugging face transformer)
I am happy today.	Joy 77.4%	Joy 75.4%	Joy 76.4%	Joy 95.2%
Alas, I lost all my project data due to a technical glitch with tacotron	Sad 75.2%	Sad 80.6%	Sad 77.9%	Sad 88.7%
It's frustrating how unreliable the results are . It's making me so angry!	Anger 97.8%	Anger 83.9%	Anger 90.9%	Anger 98.9%
I'm scared of what the future holds.	Fear 99.2%	Fear 99.8%	Fear 99.5%	Fear 99%
I didn't expect you to remember my birthday.	Surprise 75.03%	Surprise 72.24%	Surprise 73.6%	Surprise 61.8%
I feel so embarrassed about what I did.	Shame 99.29%	Shame 99.8%	Shame 99.5%	Embarrassment 99.2%

TABLE 8.1 Emotion detection from text using Linear Regression, Support Vector Machine and EmoRoberta Model for Hugging face transformer with their confidence score.

Below shows the confusion matrix for all three models:

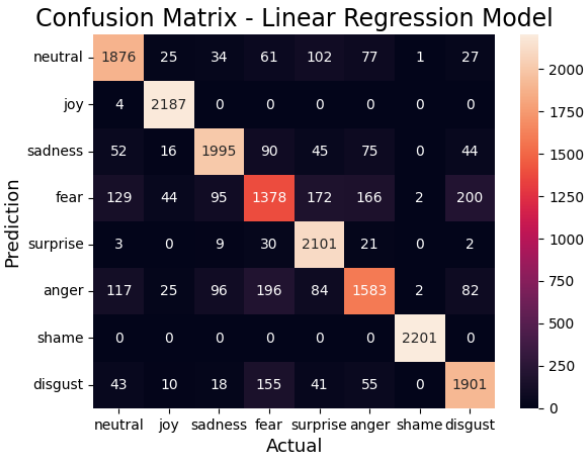

Figure 8.1.1

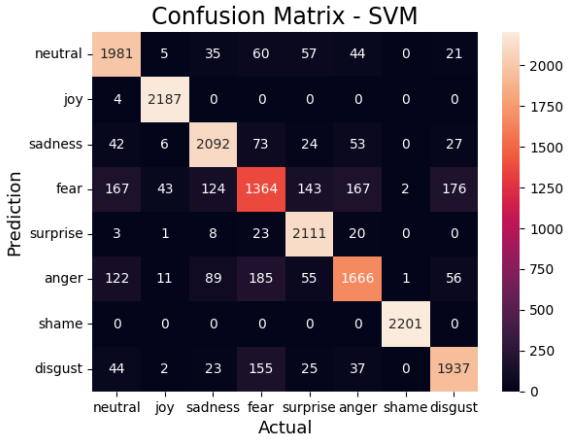

Figure 8.1.2

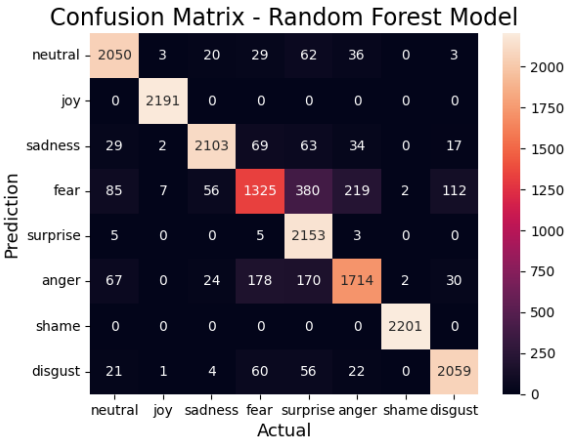

Figure 8.1.3

8.2 For the speech model

We have trained the tacotron model for more than 9000 steps and achieved the following results. The loss rate decreased over the number of steps and showed a constant decline after 9000 steps. This indicates the convergence to optimality and helps in the quality audio of the emotional speech. The same can be confirmed by the loss rate decreasing in the mel spectrogram as well which can confirm the improved quality of the emotional audio too.

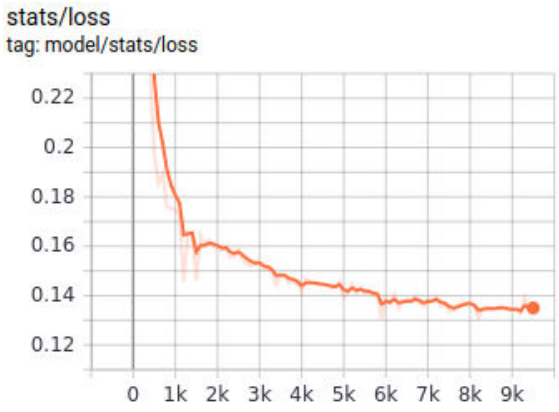

Figure 8.2.1

Figure 8.2.2

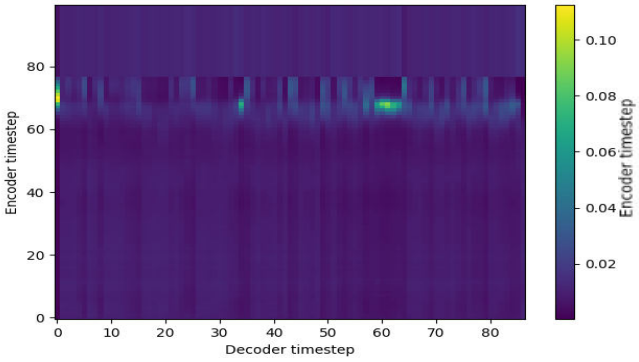

Figure 8.2.3

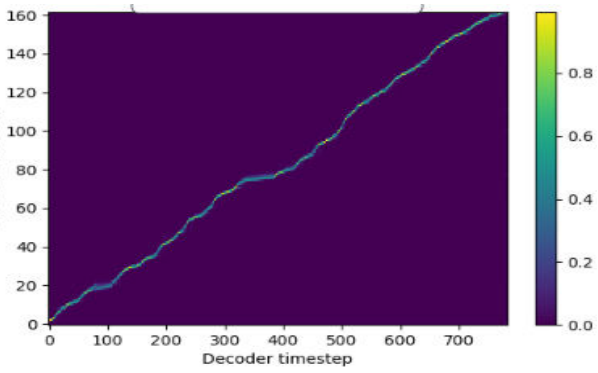

Figure 8.2.4

From the above analysis, we get to see that quality of audio has improved and notable speech with the desired emotion is observed too.

Test of the emotional speech and results:

We used a tool known as Vokaturi, which can detect the kind of emotion from the emotional speech. Here are the findings of our emotional speech against the above-mentioned tool.

Speech	Expected	Happy %	Sad %	Neutral %
"Ha Ha, this joke is so funny to laugh"	Happiness	73.2	0.5	2.31
"I am so excited for the vacation"	Happiness	81.4	10.4	1

“Alas, I lost all my project data due to a technical glitch with tacotron”	Sadness	18.1	81.0	0.4
“Oh, I feel grieved depressed about the mournful incident”	Sadness	0.26	63.1	2
“He is chasing the butterflies”	Neutral	20.2	2.3	75.4
“I am working on this project”	Neutral	10.4	15.6	78.7

TABLE 8.2 Samples of speech which has shown the corresponding confidence of emotion from the tool.

```
smsraj@sms-desktop:~/Desktop/OpenVokaturi-4-0/OpenVokaturi-4-0$ python3 examples/OpenVokaWavMean.py Sad/2.wav
Loading library...
Analyzed by: OpenVokaturi version 4.0 for open-source projects, 2022-08-22
Distributed under the GNU General Public License, version 3 or later
Reading sound file...
   sample rate 20000.000 Hz
Allocating Vokaturi sample array...
   72000 samples, 1 channels
Creating VokaturiVoice...
Filling VokaturiVoice with samples...
Extracting emotions from VokaturiVoice...
Neutral: 0.004
Happy: 0.181
Sad: 0.810
Angry: 0.001
smsraj@sms-desktop:~/Desktop/OpenVokaturi-4-0/OpenVokaturi-4-0$
```

```
smsraj@sms-desktop:~/Desktop/OpenVokaturi-4-0/OpenVokaturi-4-0$ python3 examples/OpenVokaWavMean.py Happy/3.wav
Loading library...
Analyzed by: OpenVokaturi version 4.0 for open-source projects, 2022-08-22
Distributed under the GNU General Public License, version 3 or later
Reading sound file...
   sample rate 20000.000 Hz
Allocating Vokaturi sample array...
   60000 samples, 1 channels
Creating VokaturiVoice...
Filling VokaturiVoice with samples...
Extracting emotions from VokaturiVoice...
Neutral: 0.016
Happy: 0.720
Sad: 0.067
Angry: 0.162
```

Figure 8.2.5 and 8.2.6

9. CONCLUSIONS AND FUTURE WORK

In conclusion, this paper presents a novel and impactful framework for imbuing synthesized speech with emotion, encompassing a two phase methodology of training and testing. The training phase involves the development of models for emotion detection from labeled text data and the acquisition of emotion specific pitch, intensity, and modulation from labeled speech data. Furthermore, the integration of text to speech (TTS) methods facilitates the conversion of input text to neutral speech. In the testing phase, the framework effectively leverages an emotion detection model, TTS, and a tacotron model to seamlessly apply emotionspecific speech features to the mel spectrograms obtained after passing the annotated text with emotions to the encoder resulting in emotionally expressive synthesized speech. This work underscores the potential of machine learning techniques in creating emotionally resonant audio from commonplace text inputs. The model can read not more than 300 characters of text (i.e. a normal sentence) to give out an emotional speech. The proposed framework represents a significant contribution to the field of natural language processing and TTS.

Future works could include multiple emotions, sarcasm detection, multi-speaker, reading large audiobooks with emotion, etc…

REFERENCE

[1] Ren, Yi, Ruan, Yangjun, Tan, Xu, Qin, Tao, Zhao, Sheng, Zhao, Zhou, and Tie Liu. "FastSpeech: Fast, Robust and Controllable Text to Speech." ArXiv, (2019).

[2] Liu, Rui, et al. "Modeling prosodic phrasing with multitask learning in tacotron based TTS."

[3] Berrak Sisman, Member, IEEE, Guanglai Gao, Haizhou Li, Fellow, IEEE, 2021, “Expressive TTS Training with Frame and Style Reconstruction-Loss”

[4] P. Chandra et al., "Contextual Emotion Detection in Text using Deep Learning and Big Data," 2022

BIBLIOGRAPHY

Voice_datasets : Used for emotional speech datasets for training the tacotron model. the datasets used are LJ Speech and IEMOCAP

EmoRoberta Model from Hugging Transformers: RoBERTa builds on BERT’s language masking strategy and modifies key hyperparameters in BERT, including removing BERT’s next-sentence pre training objective, and training with much larger mini-batches and learning rates.

Emotion Detection from Text dataset: Used for train linear regression, SVM and Random Forest models to detect emotion from text.

Hugging face: Consist of pre-built AI models.

LeXmo: LeXmo is a python package for emotion classification.

Vokaturi tool : Used for testing confidence of all the emotions in the emotional speech

APPENDIX A : DEFINITIONS, ACRONYMS AND ABBREVIATIONS

Emotion detection model: A model that analyzes text or speech and identifies the corresponding emotion, such as anger, sadness, happiness, or sarcasm.

Preprocessing: The process of cleaning and preparing input data for further processing.

Stop words: Commonly used words that are often removed from text data during preprocessing.

Explicit words: Words that may be inappropriate or offensive in certain contexts.

Punctuation: Marks such as periods, commas, and quotation marks that are used to separate and structure text.

Deep learning: A subset of machine learning that involves training neural networks to perform complex tasks.

Annotated text: Text that includes additional information, such as labels or tags, to provide context and meaning.

Role Based dialogue format: A format of text that is used in a dialogue between two or more people, with each person assigned a specific role.

Prosody: The patterns of stress and intonation in speech that convey emotional content.

Mel spectrogram: It applies a frequency-domain filter bank to audio signals that are windowed in time.

N- grams: contiguous sequences of n items such as words or characters, from a given sample of text or speech.

Linear Regression model: A statistical method that models the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data.

GriffinLim algorithm: A phase reconstruction technique used in audio signal processing, particularly in the context of audio source separation, to estimate the missing phase information of a signal from its magnitude spectrogram.

Hyperparameters: External configuration settings for a machine learning model that are not learned from the data but are set prior to the training process, influencing the model's behavior and performance.

Sequence of Phonemes: Consecutive series of distinct speech sounds, which are the basic units of sound in a language, often used for phonetic or linguistic analysis.

SER network - A Speech Emotion Recognizer (SER) network is a type of artificial intelligence (AI) model that is designed to identify and classify the emotional content of human speech.

Tacotron model - Tacotron is an end-to-end text-to-speech (TTS) system that uses a sequence-to-sequence (S2S) model to learn the mapping between text and speech.

Seq2Seq - Seq2seq: , an abbreviation for "sequence to sequence," is a family of machine learning models for processing sequences, particularly for tasks involving the transformation of one sequence to another sequence.

FastSpeech - FastSpeech is a non-autoregressive neural network-based text-to-speech (TTS) model that offers high-quality speech synthesis while significantly reducing inference latency compared to traditional autoregressive TTS models.

FeedForward Neural network - A feedforward neural network (FNN) is a type of artificial neural network (ANN) where connections between nodes do not form a loop.

Emotion Annotation: Expanding the annotation process to encompass a wider range of emotional expressions in text.

Tacotron Enhancement: "Upgrading Tacotron model architecture with emotion-aware components for nuanced speech synthesis.

NLP: Natural Language Processing

LSTM: Long ShortTerm Memory

CNN: Convolutional Neural Network

RNN: Recurrent Neural Network

TTS: Text to Speech

SVM: Support Vector Machine

“FeelSpeak: Generating Emotional Speech with Deep Learning”

Word Count: 7956

Plagiarism Percentage 14%

Matches

1 **World Wide Web Match**
View Link

2 **World Wide Web Match**
View Link

3 **World Wide Web Match**
View Link

4 **World Wide Web Match**
View Link

5 **World Wide Web Match**
View Link

6 **World Wide Web Match**
View Link

7 **World Wide Web Match**
View Link

8 **World Wide Web Match**
View Link

9 **World Wide Web Match**
View Link

10 **World Wide Web Match**
View Link

11 **World Wide Web Match**

View Link

12 **World Wide Web Match**
View Link

13 **World Wide Web Match**
View Link

14 **World Wide Web Match**
View Link

15 **World Wide Web Match**
View Link

16 **World Wide Web Match**
View Link

17 **World Wide Web Match**
View Link

18 **World Wide Web Match**
View Link

19 **World Wide Web Match**
View Link

20 **World Wide Web Match**
View Link

21 **World Wide Web Match**
View Link

22 **World Wide Web Match**
View Link

23 **World Wide Web Match**
View Link

24 **World Wide Web Match**
View Link

25 **World Wide Web Match**
View Link

26 **World Wide Web Match**
View Link

27 **World Wide Web Match**
View Link

28 **World Wide Web Match**
View Link

29 **World Wide Web Match**
View Link

30 **World Wide Web Match**
View Link

31 **World Wide Web Match**
View Link

32 **World Wide Web Match**
View Link

33 **World Wide Web Match**
View Link

34 **World Wide Web Match**
View Link

35 **World Wide Web Match**
View Link

36 **World Wide Web Match**
View Link

37 **World Wide Web Match**
View Link

38 **World Wide Web Match**

View Link

39 **World Wide Web Match**
View Link

40 **World Wide Web Match**
View Link

41 **World Wide Web Match**
View Link

42 **World Wide Web Match**
View Link

43 **World Wide Web Match**
View Link

44 **World Wide Web Match**
View Link

45 **World Wide Web Match**
View Link

46 **World Wide Web Match**
View Link

47 **World Wide Web Match**
View Link

Suspected Content

Dissertation on “FeelSpeak: Generating Emotional Speech with Deep Learning”

Submitted in partial fulfilment of the requirements for the award of degree of 12
Bachelor of Technology in Computer Science & Engineering

UE20CS390B – Capstone Project Phase 2 Submitted by: M H SOHAN RAHUL ROSHAN G ROHIT ROSHAN S M SUTHARSAN RAJ PES1UG20CS235 PES1UG20CS320 PES1UG20CS355 PES1UG20CS362 Under the guidance of Prof. V R BADRI PRASAD Associate Professor June December 2023