SPEECH EMOTION DETECTION USING CNN AND LSTM

Team 1

CST461 DEEP LEARNING

12/08/2024

NIHAR JANI

PARTH OZA

TOCHUKWU EZEKWERE

Abstract:

This project involves the use of Long Short Term Memory Networks(LSTMs) and CNN to predict the emotion exuded from a line of speech. The model was trained using the The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS). The dataset contained 7356 audio files of 24 professional actors (12 female, 12 male), vocalizing two lexically-matched statements in a neutral North American accent. The speech in the files includes calm, happy, sad, angry, fearful, surprise, and disgust emotions.

Introduction:

Speech emotion detection using CNN and LSTM is a project aimed at developing a robust system for recognizing emotions in spoken language. The context and objectives of this project are:

1. To address the growing importance of emotion recognition in human-computer interaction and digital healthcare.
2. To develop a model capable of accurately classifying emotions such as happiness, sadness, anger, and fear from speech signals.
3. To leverage the strengths of both Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks for improved performance
4. To extract relevant acoustic features from speech signals using signal processing techniques like Fourier Transform and Mel-frequency cepstral coefficients (MFCCs)
5. To create a system that can analyze emotions in real-time scenarios, emphasizing practical applications
6. To contribute to advancements in fields such as sentiment analysis, virtual assistants, and emotion-aware systems
7. To explore the potential application of this technology in managing mental health conditions like depression and anxiety within digital healthcare.

The project aims to achieve high accuracy in emotion classification by combining the feature extraction capabilities of CNNs with the temporal modeling strengths of LSTMs, ultimately

creating a powerful tool for understanding human emotions in spoken language.

Methodology:

Speech Emotion Recognition using CNN: Methodology Summary

This project implements a Speech Emotion Recognition system using a Convolutional Neural Network (CNN) trained on the **RAVDESS dataset**. The methodology involves the following steps:

1. Data Preprocessing

- **Dataset Selection:** Audio files are sourced from the RAVDESS dataset, which includes labeled emotions such as *happy*, *sad*, *angry*, and *neutral*.
- **Feature Extraction:** Audio signals are processed to extract features using librosa:
  - **MFCCs:** Captures audio spectrum shape.
  - **Chroma:** Encodes harmonic content.
  - **MEL Spectrogram:** Models perceptual sound characteristics.
  - **Contrast and Tonnetz:** Additional spectral features.
- The extracted features are saved as numerical vectors for further processing.

2. Data Splitting

- The dataset is divided into **training (75%)** and **testing (25%)** subsets using train_test_split from sklearn.
- Features and labels are converted to NumPy arrays and reshaped to include a single channel dimension for compatibility with CNN input requirements.

3. Model Architecture

A Sequential CNN model is designed with the following layers:

1. **First Convolutional Layer:**
   a. 128 filters, kernel size of 5, ReLU activation, followed by Dropout (10%) and MaxPooling (pool size = 8).
2. **Second Convolutional Layer:**
   a. Similar to the first, without MaxPooling.
3. **Flattening Layer:**
   a. Converts 2D feature maps into 1D vectors.
4. **Dense Output Layer:**
   a. Fully connected with 8 neurons for 8 emotions, using softmax activation.

4. Model Compilation

- **Loss Function:** Sparse categorical cross-entropy for multi-class classification.
- **Optimizer:** RMSProp with a learning rate of 0.00005.
- **Evaluation Metric:** Accuracy.

5. Model Training

- The model is trained over **500 epochs** with a batch size of 20, using the testing data for validation.
- Training performance metrics, such as accuracy and loss, are tracked during each epoch.

6. Prediction and Mapping

- A mapping function is implemented to convert numerical predictions back to emotion labels (*e.g., 0 → happy*).

This pipeline provides an efficient approach to recognizing emotions from speech using deep learning techniques.

Summary of Methodology: Speech Emotion Recognition Using LSTM

Objective

To develop a robust system for recognizing emotions from speech using audio feature extraction and an LSTM-based neural network.

1. Data Acquisition

- **Dataset**: Audio files from a speech emotion recognition dataset (RAVDESS).
- **Structure**: Files are categorized by emotion, encoded within filenames.
- **Label Mapping**: Integer labels (e.g., 1 for neutral) are converted to descriptive names (e.g., neutral, happy, etc.).

2. Data Preprocessing

- **Loading**: Audio files are loaded using librosa, with parameters to handle silence and ensure uniform length (e.g., 2.5 seconds with a 0.6-second offset).
- **Augmentation**:
  - **Noise Injection**: Adds random noise to simulate real-world variations.
  - **Time Stretching**: Modifies speed without altering pitch.
  - **Pitch Shifting**: Changes pitch to create diversity in training data.

3. Feature Extraction

- **MFCCs (Mel-Frequency Cepstral Coefficients)**:
  - Extracted from each audio sample as they represent essential features for speech patterns.
  - Processed features are framed into a fixed size (e.g., 108 time steps) to match the model's input requirements.

4. Model Architecture

- **Base Framework**: Built using the Keras library.
- **Layers**:
  - **LSTM Layers**: Handle sequential dependencies in audio data.
  - **Dense Layers**: Perform classification on extracted features.
  - **Batch Normalization**: Stabilizes learning by normalizing layer inputs.
  - **Dropout**: Reduces overfitting by randomly dropping connections during training.
- **Input Reshaping**: Features are expanded and swapped to align with LSTM input format ([samples, timesteps, features]).

5. Training and Evaluation

- Data is split into training and test sets using train_test_split.
- **Loss Function**: Categorical cross-entropy for multi-class classification.
- **Metrics**: Accuracy, confusion matrix, and classification report.
- **Optimization**: Adam optimizer is employed for adaptive learning rates.

6. Output

- Predicted emotion classes for input speech samples.
- Performance is evaluated using precision, recall, F1-score, and overall accuracy.

This methodology combines advanced audio preprocessing, feature extraction, and an LSTM-based architecture to achieve robust speech emotion recognition.

Results:

CNN

```
...                precision    recall  f1-score   support

           0.0       0.75      0.81      0.78        94
           1.0       0.80      0.89      0.85       101
           3.0       0.78      0.64      0.70        44
           4.0       0.94      0.83      0.88        90

      accuracy                           0.82       329
     macro avg       0.82      0.79      0.80       329
  weighted avg       0.82      0.82      0.82       329

  [[76 10  3  5]
   [ 6 90  5  0]
   [ 5 11 28  0]
   [14  1  0 75]]
```

```
    loss, acc = sm.evaluate(x_testcnn, y_test)
    print("Restored model, accuracy: {:5.2f}%".format(100*acc))
```

```
...  329/329 [==============================] - 0s 130us/step
  Restored model, accuracy: 81.76%
```

LSTM

	precision	recall	f1-score	support
angry	0.86	0.88	0.87	130
calm	0.75	0.84	0.80	120
disgust	0.87	0.83	0.85	105
fear	0.61	0.74	0.67	114
happy	0.60	0.63	0.61	102
neutral	0.53	0.60	0.56	62
sad	0.64	0.44	0.52	108
surprise	0.92	0.80	0.85	114
accuracy			0.73	855
macro avg	0.72	0.72	0.72	855
weighted avg	0.74	0.73	0.73	855

Discussion:

The search results reveal details about a speech emotion recognition project using machine learning models. Here are the key points and their implications:

Dataset and Emotions

The project uses the RAVDESS dataset, which includes audio files with eight emotions:

1. Neutral
2. Calm
3. Happy
4. Sad
5. Angry
6. Fearful
7. Disgust
8. Surprised

However, the project focuses on a subset of four emotions: angry, sad, neutral, and happy. This simplification has several implications:

- Reduced complexity, potentially improving model accuracy for these core emotions
- Faster training and inference times
- Limited applicability in scenarios requiring a broader range of emotion detection

Feature Extraction

The code extracts various audio features for emotion recognition

- MFCC (Mel-Frequency Cepstral Coefficients)
- Chroma
- Mel Spectrogram
- Contrast
- Tonnetz

Model Architecture

Two different model architectures:

1. LSTM (Long Short-Term Memory)
2. CNN (Convolutional Neural Network)

The use of these advanced neural network architectures implies:

- Ability to capture temporal dependencies in speech (LSTM)
- Effective feature learning from spectral representations of audio (CNN)
- Potential for high accuracy in emotion classification

Data Processing and Model Training

The code includes functions for:

1. Loading and processing audio files

2. Extracting features

3. Splitting data into training and testing sets

This structured approach ensures:

- Consistent data preparation across the dataset

- Proper evaluation of model performance

- Reproducibility of results

Implications and Applications

1. Mental Health Monitoring: The system could be used to track emotional states over time, potentially aiding in the diagnosis and treatment of mood disorders.

2. Customer Service Enhancement: Automated systems could analyze customer emotions during calls, allowing for better response tailoring and service quality assessment.

3. Human-Computer Interaction: Emotion-aware interfaces could adapt their behavior based on the user's emotional state, improving user experience.

4. Limited Emotion Range: By focusing on four emotions, the model may have limitations in real-world scenarios where a broader range of emotions is present.

5. Privacy Concerns: The ability to detect emotions from speech raises questions about privacy and consent in recorded conversations.

6. Cross-cultural Applicability: Emotion expression can vary across cultures, so the model's performance may differ depending on the cultural context.

Conclusion and Future Work:

This speech emotion recognition project makes several key contributions and opens up potential areas for further research:

Contributions

1. Dataset Utilization: The project uses the RAVDESS dataset, which contains audio files with eight distinct emotions. This comprehensive dataset allows for a robust analysis of emotional speech patterns

2. Feature Extraction: The project employs a diverse set of audio features for emotion recognition, including MFCC, Chroma, Mel Spectrogram, Contrast, and Tonnetz. This multi-feature approach likely enhances the model's ability to capture subtle emotional cues in speech

3. Model Architecture: The project explores two advanced neural network architectures:

- LSTM (Long Short-Term Memory): Capable of capturing temporal dependencies in speech

- CNN (Convolutional Neural Network): Effective for learning features from spectral representations of audio

4. Emotion Subset: By focusing on four core emotions (angry, sad, neutral, happy), the project simplifies the classification task while still covering a range of fundamental emotional states

Potential Areas for Further Research

1. Expanded Emotion Range: Future work could explore incorporating the full range of emotions from the RAVDESS dataset, potentially improving the model's real-world applicability.

2. Cross-cultural Analysis: Investigating how the model performs across different languages and cultural contexts could lead to more universally applicable emotion recognition systems.

3. Multi-modal Emotion Recognition: Integrating visual cues (facial expressions, body language) with audio features could enhance the accuracy of emotion detection.

4. Real-time Processing: Adapting the model for real-time emotion recognition could open up applications in live customer service or mental health monitoring.

5. Transfer Learning: Exploring how well the model transfers to other speech-related tasks or datasets could reveal its generalizability.

6. Explainable AI: Developing methods to interpret the model's decision-making process could increase trust and understanding of its predictions.

7. Emotion Intensity: Extending the model to not only classify emotions but also measure their intensity could provide more nuanced emotional analysis.

References:

1. RAVDESS Dataset: Ryerson Audio-Visual Database of Emotional Speech and Song. This dataset is used in both projects for speech emotion recognition.

2. Librosa: McFee, B., et al. (2015). librosa: Audio and Music Signal Analysis in Python. Proceedings of the 14th Python in Science Conference.

3. Keras: Chollet, F., et al. (2015). Keras. GitHub repository: https://github.com/fchollet/keras

4. SciKit-Learn: Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

5. Speech Emotion Recognition using LSTM: Reference to the Jupyter Notebook "SpeechEmotionRecognition_model_LSTM.ipynb"

6. Speech Emotion Recognition using CNN: Reference to the Jupyter Notebook "Speech-Emotion-Recognition-using-CNN.ipynb"

7. Feature Extraction Techniques:

- MFCC (Mel-Frequency Cepstral Coefficients)
- Chroma
- Mel Spectrogram
- Spectral Contrast
- Tonnetz