EEG-based Emotion Recognition via Channel-wise Attention and Self Attention

Problem

- Most methods extract discriminative features and ignore useful information in channel and time.

Previous works

Conti-CNN

- combine the features of multiple bans to improve recognition accuracy

GCNN

- adopt different entropy (DE) feature as inputs, and use the spectral graph filtering to extract features and recognize emotion

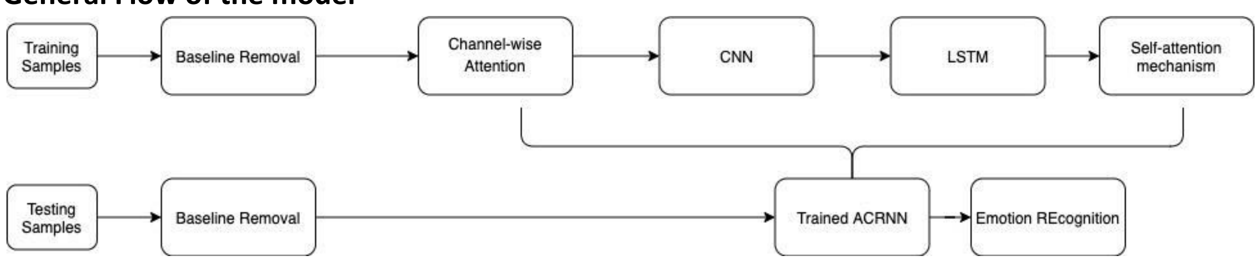
CRAM

- utilize a CNN to encode the high-level representation of EEG signals and a recurrent attention mechanism to explore the temporal dynamics.

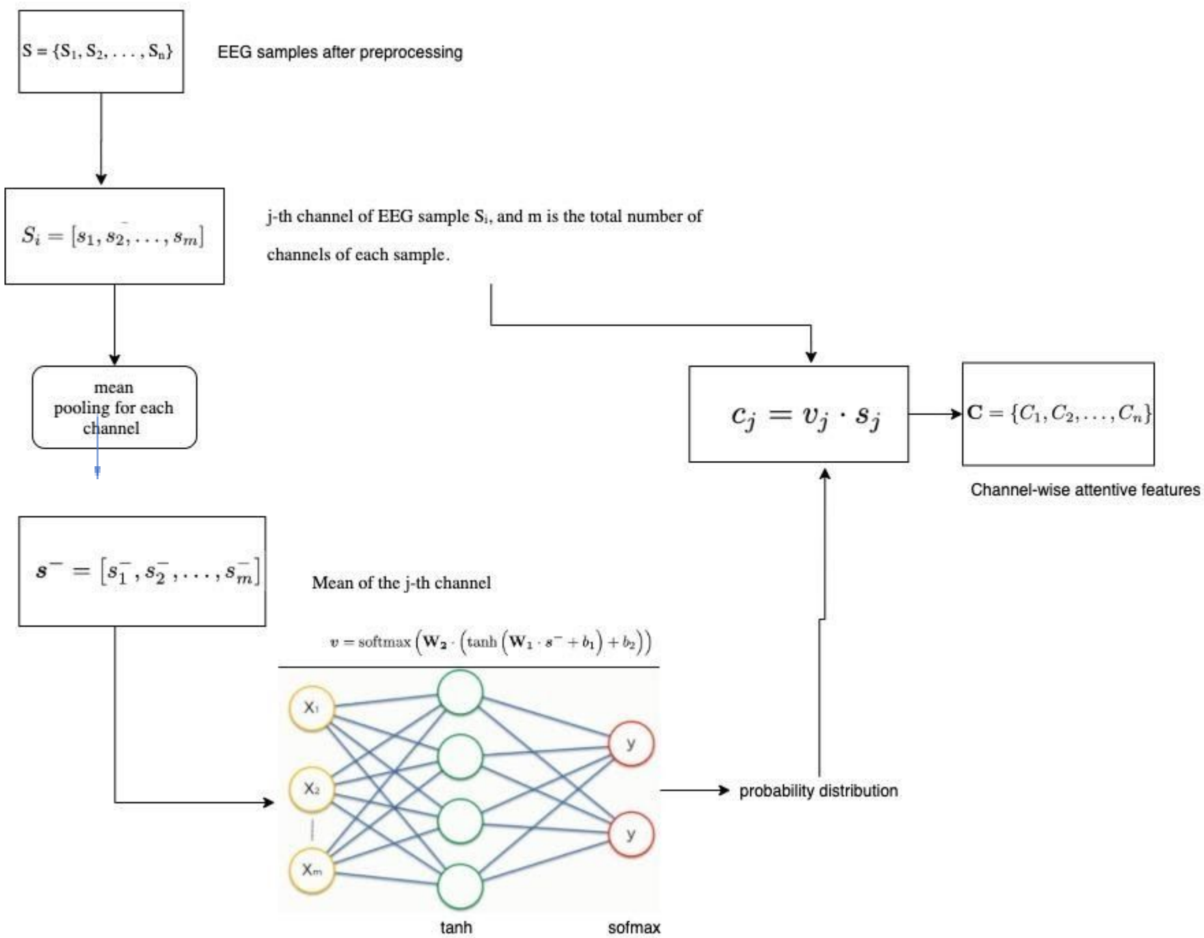
Idea

- an attention-based convolutional recurrent neural network (ACRNN) to extract more discriminative features from EEG signals and improve the accuracy of emotion recognition.
    - Channel-wise attention mechanism for CNN
    - Self-attention mechanism for RNN

General Flow of the model



Channel-wise Attention



- extract the difference among channels from the EEG signals by assigning the weights to different channels.
- change the weight of different channels to explore the information of a feature map
- squeeze the global spatial information and generate channel-wise statistics
- two fully-connected (FC) layers around the non-linearity
- softmax function transforms the importance of channels to probability distribution v
    - v: importance of different channels.
- we consider probability as the weight to recode the information of the EEG sample S

CNN

em

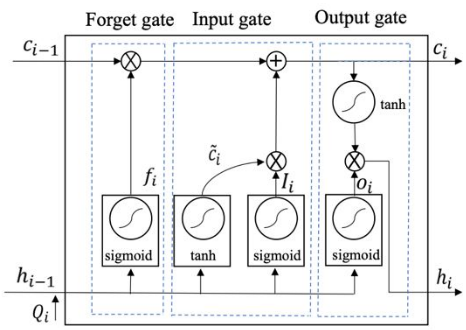
- the kernel height is the same as the number of electrodes.
- the kernel width is also designed to explore temporal information of the EEG signals
- use the exponential linear unit (ELU) function as the activation function in the convolution operations

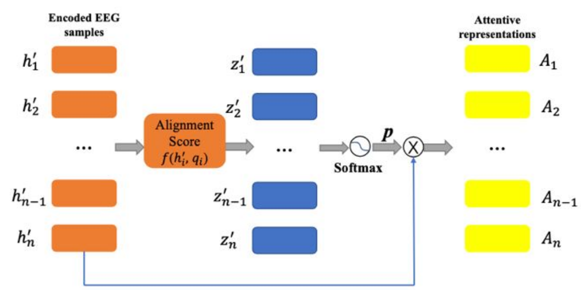
LSTM

- the LSTM cell exports two outputs, i.e., output ci at the current time i and hidden state hi
- the number of LSTM units in each layer is the same as the number of EEG samples

- two stacked layers to remember and encode all scanned spatial and temporal areas

Self-attention mechanism





- assign weights to each EEG signal sample by exploring the intrinsic importance of each sample.
- compute the similarity within each sample from different
  - $$z'_i = f\left(h'_i, q_i\right) = W^T \sigma\left(W_1 h'_i + W_2 q_i + b_1\right) + b,$$
    - f (h′ i , qi) represents the intrinsic similarity of the i-th encoded EEG sample
    - qi is the aligned pattern vector generated based on the feature vector h'_i by linear trans- formation
  - activation functions: ELU
  - W and b are the weight and bias terms of σ function,
  - P: the probabilities of all samples
    - the probability of the i-th EEG sample

$$p_i = \frac{\exp\left(z_i'^T \cdot h'_i\right)}{\sum_{i=1}^{n} \exp\left(z_i'^T \cdot h'_i\right)}.$$

  - A: the features ex- tracted by the extended self-attention mechanism
    - the i-th attentive feature

$$A_i = p_i \cdot h'_i.$$

  - Softmax layer:

$$\boldsymbol{P} = \operatorname{softmax}(W\boldsymbol{A} + b),$$

  - cross-entropy error

$$\mathcal{L} = -\sum_{i=1}^{n} \hat{Y}_i \log\left(P_i\right),$$

- Y_i is the label of the i-th EEG sample
    - the lower cross-entropy error L indicates higher emotion recognition accuracy.

Emotion Recognition from Multi-Channel EEG through Parallel Convolutional Recurrent Neural Network

problem

- directly employ the EEG signals without taking into account the role of the baseline (EEG signals without stimulation).
- rely on complex pre-processing and hand-engineered features to a great extent,

solution

1. take the baseline signals into account
2. transform the raw 1D chain-like EEG signals into 2D frame-like sequences.
    a. signals come from physically adjacent channels are still adjacent in the frame,
        i. reason: the spatial information can be retained after converting
    b. a hybrid deep learning structure that integrates the Convolutional Neural Network and Recurrent Neural Network to conduct emotion recognition tasks
        i. CNN: extract spatial features from data frames.
        ii. RNN: extract temporal features from EEG sequence.
    c. a feature fusion method is applied to fuse the spatial features and temporal features.

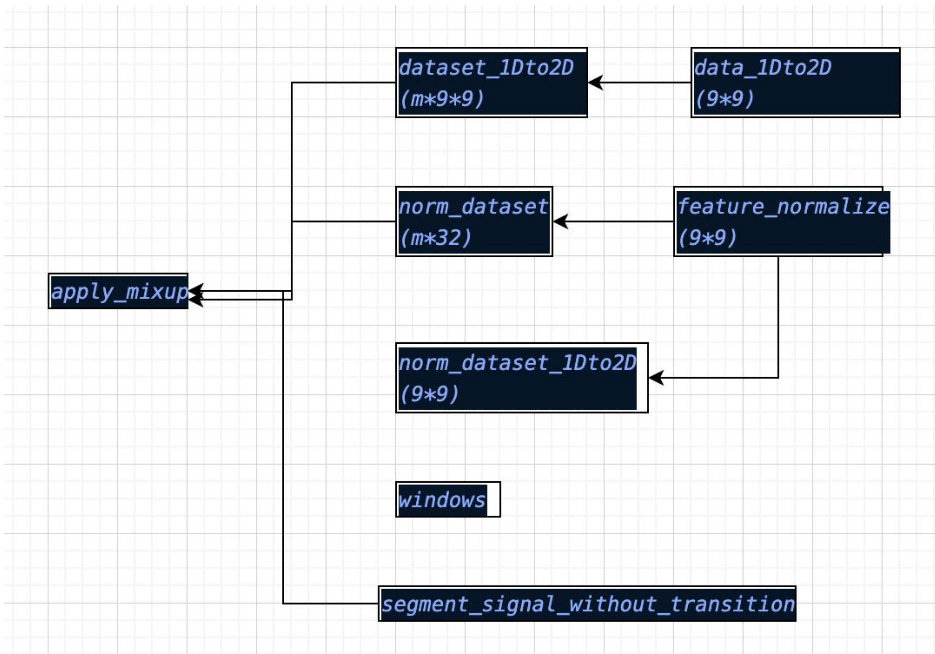
Dataset - DEAP

Array name	*array shape*	*Array contents*
data	40 × 40 × 8064	video/trial × channel × data
labels	40×4	video/trial × label (valence, arousal, dominance, liking)

Preprocessing

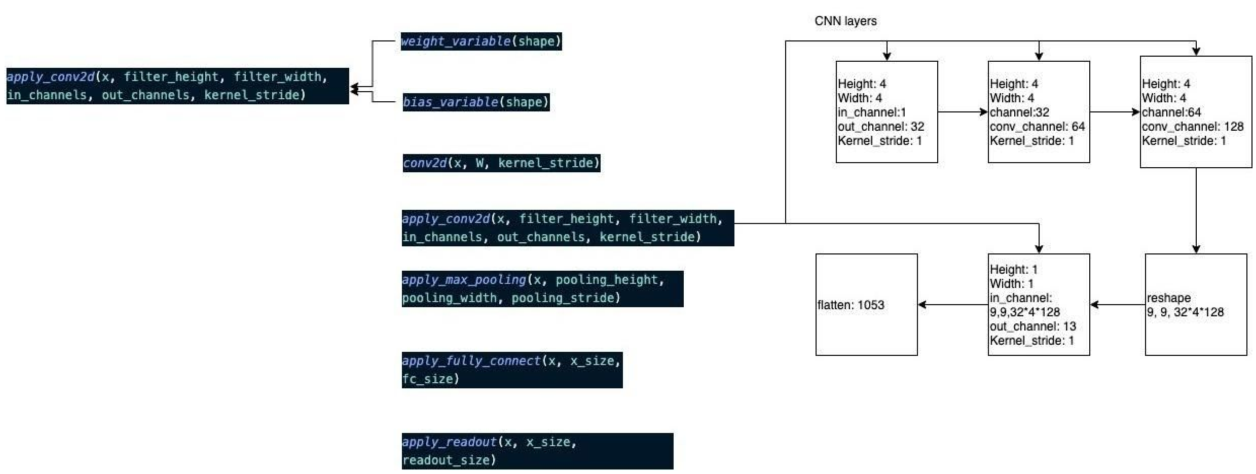
- take out pre-trial signals from all C channels and cut it into N segments with a same length L.
    - C: channels
    - N: segments
    - L: length of each segment
    - N(C*L)
- 60s trial data; 3s baseline data
- $$\text{BaseMean} = \frac{\sum_1^N mat_i}{N}$$
- Pre-trial - Baseline
    - $$\text{BaseRemoved}_j = \text{RawEEG}_j - \text{BaseMean}$$

- Detailed function



CNN

- Use Z-score normalization
- The CNN unit works for mining cross-channel correlation and extracting features from 2D frames.
- Use 4*4 filter
    - 4 by 4 filter can mine the correlation among more channels than 3 by 3 kernel
- use zero-padding to prevent missing information at the edge of input data frame.
- first convolutional layer with 32 feature maps and double the feature maps in each of the following convolutional layers.
- The pooling layer is usually added for reducing data dimensional at the cost of missing some information.
- a batch normalization (BN) operation is applied to accelerate the model training.



Results

Recognition Accuracy (%) Comparison for Each Subject on "Arousal"		
Sub	results	Given Results
1	93.87%	93.00%
2	85.97%	86.68%
3	94.47%	95.45%
4	85.58%	84.78%
5	89.35%	88.40%
6	88.35%	90.10%
7	90.98%	90.68%
8	91.25%	92.55%
9	88.75%	88.35%
10	91.23%	89.85%

Recognition Accuracy (%) Comparison for Each Subject on "Valence"		
Sub	results	Given Results
1	92.92%	92.93%
2	84.07%	85.07%
3	91.57%	94.80%
4	84.53%	85.42%

A Principled Approach for Learning Task Similarity in Multitask Learning

What problem does Multitask Learning solve

- understand the similarities within a set of tasks.
    - Two approaches
        - incorporated this similarity information explicitly
            - weighted loss for each task
        - incorporated this similarity information Implicitly
            - adversarial loss for feature adaptation

Previous Works

[Wang and Pineau, 2015]

- In the multitask learning (MTL) scenario, an agent learns the shared knowledge between a set of related tasks.

[Murugesan and Carbonell, 2017; Murugesan et al., 2016; Pentina and Lampert, 2017]

- minimize a weighted sum of empirical loss in which similar tasks are assigned higher weights.

[Liu et al., 2017; Li et al., 2018]

- use adversarial losses by feature adaptation, minimizing the distribution distance between the tasks to con- struct a shared feature space.