language: en license: cc-by-4.0

tags:

- text-classification
- evidence-detection

Model Card for b64065ab-c62354yt-ED

This is a binary classification model used for an evidence detection task. This model is based on the BiLSTM architecture.

Model Details

Model Description

This model is based upon a BiLSTM architecture that was trained on 23,000 labelled training examples.

- **Developed by:** Anandajyothis Binu and Yi Jun Tew
- **Language(s):** English
- **Model type:** Supervised
- **Model architecture:** BiLSTM

Training Details

Training Data

23K labelled datapoints for this task was provided for training.

Training Procedure

The preprocessing step involves tokenising both claim and evidence sequences. The sequences then padded/truncated to fit a specified length. This length was 13 for claim sequences and 65 for evidence sequences. These values were chosen because the 95th percentile claim and evidence legnths are below these values.

Training Hyperparameters

```
- batch_size: 16
- num_epochs: 4
- num_bilstm_layers: 3
- dropout_0: 0.2
- droupout_1: 0.0
- droupout_2: 0.0
- lstm_units_0: 64
- lstm_units_1: 32
- lstm_units_2: 32
```

```
- num_dense_layers: 1
- dense_units_0: 32
- dense_dropout_0: 0.4
- learning_rate: 1e-3
```

Speeds, Sizes, Times

```
- overall training time: 15-40 minutes
- duration per training epoch: 3-10 minutes
- model size: 16.5MB
```

Evaluation

Testing Data & Metrics

Testing Data

A subset of the development set provided, accounting for 6K datapoints, were used as the validation set.

Metrics

```
- Accuracy
- Precision
- Recall
- F1-score
```

Results

The model obtained an F1-score of 83% and an accuracy of 83%.

Precision: 82% Recal: 83%

Technical Specifications

Hardware

```
- RAM: at least 4 GB
- Storage: at least 100MB
- GPU: optional
```

Software

```
- TensorFlow 2.15.0
- keras_tuner 1.4.7
```

Bias, Risks, and Limitations

All claim sequences are truncated after 13 tokens and padded to fit this dimension. A 65 token limit is applied to the evidence sequences. These values were chosen to accomodate for the 95th percentile of sequence lengths. Any tokens after these truncation limits will not be considered by the model.

Additional Information

The hyperparameters were trained using keras_tuner. Keras Tuner searched the given hyperparameter space using the hyperband strategy. This helped prune seemingly suboptimal values for hyperparameters. The evaluation metric used was classification accuracy on the validation set.