Why and when should you pool? Analyzing Pooling in Recurrent Architectures
Outline

• Background on LSTMs, Pooling and Gradient Propagation
• Max-attention
• Vanishing Gradients and Training Saturation
• Positional Biases and their Extent
  • Evaluating Natural Biases
  • Learning to Skip Unimportant Words
  • Normalized Word Importance
• Conclusions
Background on LSTMs, Pooling and Gradient Propagation

LSTMs, Pooling and Attention
Pooling enhances task accuracy of BiLSTMs and helps learn better syntactic properties.

| Model        | dim | NLI dev | NLI test | Transfer micro | Transfer macro |
|--------------|-----|---------|----------|----------------|----------------|
| LSTM         | 2048| 81.9    | 80.7     | 79.5           | 78.6           |
| GRU          | 4096| 82.4    | 81.8     | 81.7           | 80.9           |
| BiGRU-last   | 4096| 81.3    | 80.9     | 82.9           | 81.7           |
| BiLSTM-Mean  | 4096| 79.0    | 78.2     | 83.1           | 81.7           |
| Inner-attention | 4096| 82.3    | 82.5     | 82.1           | 81.0           |
| HConvNet     | 4096| 83.7    | 83.4     | 82.0           | 80.9           |
| BiLSTM-Max   | 4096| **85.0**| **84.5** | **85.2**       | **83.7**       |

[Conneau et. al, 2017]
Prior Work observed gradient norms at the first hidden state for LSTMs explode for classification tasks (left) and are unstable for Addition Tasks (right).

[Review of Literature – Vanishing gradients]

[Background on LSTMs, Pooling and Gradient Propagation]

[Zhang et. al., 2018]
Alternate work posits that gradient vanishing increases as training progresses in LSTMs.

[Arjovsky et al., 2016]
Outline

• Background on LSTMs, Pooling and Gradient Propagation
• Max-attention
• Vanishing Gradients and Training Saturation
• Positional Biases and their Extent
  • Evaluating Natural Biases
  • Learning to Skip Unimportant Words
  • Normalized Word Importance
• Conclusions
Max-attention

- Generate a sentence-specific **local query** vector to calculate attention weights.
- Using max-pooled representation as a query for attention allows for a second round of aggregation among important hidden states.

\[
q^i = \max_{t \in (1,n)} (h^i_t); \quad \hat{h}_t = h_t/\|h_t\|
\]

\[
\alpha_t = \frac{\exp(\hat{h}_t^\top q)}{\sum_{j=1}^{n} \exp(\hat{h}_j^\top q)}; \quad s_{\text{emb}} = \sum_{t=1}^{n} \alpha_t h_t
\]
Outline

• Background on LSTMs, Pooling and Gradient Propagation
• Max-attention
• Vanishing Gradients and Training Saturation
• Positional Biases and their Extent
  • Evaluating Natural Biases
  • Learning to Skip Unimportant Words
  • Normalized Word Importance
• Conclusions
Vanishing Gradients

- The gradient norm across different word positions after training for 500 examples.
- BiLSTM_{LOWF} suffers from extreme vanishing gradient, with the gradient norm in the middle nearly $10^{-10}$ times that at the ends.
- Gradient propagation in pooling-based models is invariant of word position.
How does gradient vanishing change as we train our models for more epochs?

**Training saturation**

i. BiLSTM gradient vanishing recovers slowly with more epochs. Pooling methods don’t face gradient vanishing even in the initial iterations.

ii. By the time the vanishing ratios settle, the training loss is already very low leading to no more updates to the learned weights.

![Graphs showing vanishing ratios and accuracy over training batches for BiLSTM and Max-attention models.](image-url)
Outline

• Background on LSTMs, Pooling and Gradient Propagation
• Max-attention
• Vanishing Gradients and Training Saturation
• Positional Biases and their Extent
  • Evaluating Natural Biases
  • Learning to Skip Unimportant Words
  • Normalized Word Importance
• Conclusions
Positional Biases

1. Can naturally trained recurrent models skip over unimportant words in the beginning or the end of the sentence?

2. How well can different models be trained to skip unrelated words?

3. How does the position of a word impact its importance in the final prediction by a model?
Changing Test-time Distribution

Evaluating Natural Positional Biases

- Append varying amounts of random Wikipedia words to the original data at test time.
- Adding Wikipedia words to just one end does not effect BiLSTM accuracy significantly.
- As Wikipedia words added to both ends ↑, model accuracy ↓ significantly for BiLSTM.

![Graph showing test accuracy across different percentage of Wikipedia words](image)

**Changing Test-time Distribution**

**Evaluating Natural Positional Biases**

- Append varying amounts of random Wikipedia words to the original data at test time.
- Adding Wikipedia words to just one end does not effect BiLSTM accuracy significantly.
- As Wikipedia words added to both ends ↑, model accuracy ↓ significantly for BiLSTM.
Are BiLSTMs biased towards the left/right end?

- Given less training data, BiLSTMs prematurely learn to use features from only one of the two LSTM chains.
- BiLSTM is unpResponsive to any appended tokens as long as the ‘left’ text is preserved in the 1K and 5K setting. But this bias dilutes with more training samples.
Learning to Skip Unimportant Words

- Pathological scenarios where BiLSTMs in the absence of pooling can perform no better than random guessing.
- Max-attention is the best performing model in 80% of all scenarios described in the paper.

|           | Yahoo 1K | Yahoo 2K | Yahoo 10K | Yahoo (mid) + Wiki 1K | Yahoo (mid) + Wiki 2K | Yahoo (mid) + Wiki 10K | Yahoo (right) + Wiki 1K | Yahoo (right) + Wiki 2K | Yahoo (right) + Wiki 10K |
|-----------|----------|----------|-----------|------------------------|------------------------|------------------------|--------------------------|--------------------------|--------------------------|
| BiLSTM    | 38.3 ± 4.8 | 51.4 ± 2.1 | 63.5 ± 0.6 | **12.7 ± 1.1** | **12.7 ± 1.1** | **11.4 ± 0.8** | 18.8 ± 2.5 | 37.3 ± 0.9 | 60.1 ± 1.5 |
| MEANPOOL  | 48.2 ± 2.3 | 56.6 ± 0.5 | 64.7 ± 0.6 | 31.9 ± 2.3 | 43.1 ± 2.0 | 58.5 ± 0.6 | 33.9 ± 2.1 | 43.2 ± 1.0 | 58.6 ± 0.4 |
| MAXPOOL   | 50.2 ± 2.1 | 56.3 ± 1.8 | 63.9 ± 1.1 | 33.0 ± 1.0 | 40.1 ± 1.4 | 58.4 ± 1.2 | 33.1 ± 2.5 | 41.2 ± 0.9 | 60.9 ± 1.0 |
| ATT       | 47.3 ± 2.2 | 54.2 ± 1.1 | **65.1 ± 1.5** | 39.4 ± 0.5 | 45.1 ± 1.8 | 61.5 ± 1.7 | 37.9 ± 1.4 | 47.6 ± 2.3 | 62.2 ± 0.9 |
| MAXATT    | **51.8 ± 1.1** | **57.0 ± 1.1** | **65.1 ± 1.1** | **39.6 ± 0.9** | **48.5 ± 0.6** | **62.2 ± 1.6** | **40.3 ± 1.5** | **50.1 ± 1.6** | **63.1 ± 0.7** |
Normalized Word Importance

NWI metric to calculate per-position importance of words

• Sequentially replace set of $k$ consecutive tokens by <UNK>.
• Calculate the absolute change in output probability for correct class.
• Normalize over all tokens sets in the sentence; and average over the entire corpus.

Similar to the Leave-One-Out Metric [Li et al., 2016]. But aimed at evaluating positional importance over a large corpus.
Normalized Word Importance

NWI for models trained on the IMDB dataset in the (left to right) Standard, Mid and Left Settings
Are BiLSTMs biased when sentences are short?

For short sentences (< 30 Words), the BiLSTM has higher NWI for middle words, there is still a significant importance attributed to unimportant Wikipedia words.
Outline

- Background on LSTMs, Pooling and Gradient Propagation
- Max-attention
- Vanishing Gradients and Training Saturation
- Positional Biases and their Extent
  - Evaluating Natural Biases
  - Learning to Skip Unimportant Words
  - Normalized Word Importance
- Conclusions
Conclusion

Pooling in BiLSTMs can show significant benefits in:

i. low resource settings with long input sentences
ii. when words important for the prediction are sparse or in the middle of the input

Gradient vanishing in BiLSTMs in initial iterations leads to training saturation.

BiLSTMs suffer from positional biases even in short sentences (30 words).

Pooling makes models more robust to insertions of random words on either end of the input regardless of the amount of training data.

Max-attention combines the benefits of max-pooling & attention to achieve best performance on 80% of our tasks.