Seq2Seq2Sentiment: Multimodal Sequence to Sequence Models for Sentiment Analysis

Hai Pham*¹, Thomas Manzini*¹, Paul Pu Liang², Barnabás Póczos²

¹Language Technologies Institute, ²Machine Learning Department
School of Computer Science, Carnegie Mellon University

*Authors Contributed Equally
Introduction

- Research Problem
  - Does representing multiple modalities jointly improve sentiment prediction for the CMU-MOSI dataset?
Introduction

- Research Problem
  - Does representing multiple modalities jointly improve sentiment prediction for the CMU-MOSI dataset?

- Dataset
  - Multimodal Corpus of Sentiment Intensity and Subjectivity Analysis in Online Opinion Videos (MOSI)
    - Modalities: Video, Audio, Text Transcripts
    - 89 speakers, 93 videos split into 2199 labeled opinion segments
    - Labels for Sentiment: {-1, 1} or {-3, -2, -1, 0, 1, 2, 3}
Baseline & Metrics

● LSTM Multi Modal Baseline
  ○ Concatenates all modalities together and predicts sentiment from these modalities.
  ○ 75% Accuracy (Chen et. al.)

● Metrics
  ○ For both {-1, 1} or {-3, -2, -1, 0, 1, 2, 3} cases
    ○ Precision/Recall (Test)
    ○ F1 Score (Test)
Related Work

- Word-level Temporal Methods
  - Gu et al. ACL 2018, Chen et al. ICMI 2017

- Context-Dependent Methods
  - Poria et al. ACL 2017

- Memory-based Methods
  - Zadeh et al. AAAI 2018

- Tensor-based Methods
  - Liu et al. ACL 2018, Zadeh et al. EMNLP 2017

- Conditional Approaches

- Attention-based Methods
  - Bahdanau et al. 2014, Luong et al. 2015

See our paper for an exhaustive review of related work
Seq2Seq Modality Translation

The diagram illustrates a Seq2Seq model for modality translation. It consists of two main parts: the encoder and the decoder. The encoder processes the input modalities, represented as $X_0^A, X_1^A, \ldots, X_T^A$, and the decoder generates the output modalities, represented as $X_0^B, X_1^B, \ldots, X_T^B$. Both encoder and decoder are composed of LSTM layers, with additional embedding layers at the beginning. The modalities are color-coded for clarity: red for Modality A, blue for Modality B.
Seq2Seq Modality Translation

The diagram illustrates a Seq2Seq model for modality translation. It consists of two main components:

1. **Encoder Model**:
   - Modality A:
     - $X_0^A$ to $X_T^A$
     - LSTM layers
     - Embed layers

2. **Decoder Model**:
   - Modality B:
     - $X_0^B$ to $X_T^B$
     - LSTM layers

The model takes as input sequences from different modalities (A and B) and translates one modality into another. The sentiment label is typically a value in the range of (-1, +1).
Seq2Seq Modality Translation

Dataset Modalities
Encoder Model
Decoder Model
Hidden Representation
Target Variable
Hierarchical Seq2Seq Modality Translation

- **Modality A Encoder**
  - Embed
  - LSTM
  - LSTM
  - LSTM
  - LSTM

- **Modality B Encoder**
  - Embed
  - LSTM
  - LSTM
  - LSTM
  - LSTM

- **Modality C Encoder**
  - Embed
  - LSTM
  - LSTM
  - LSTM
  - LSTM

- **Joint Representation of Modalities A & B**

- **Joint Representation of Modalities A, B & C**

- **Decoder Models**
  - LSTM
  - LSTM
  - LSTM

- **Dataset Modalities**
  - Encoder Model
  - Decoder Model
  - Hidden Representation
  - Target Variable

- **Sentiment**
  - (-1, +1)
  - Dense
Experiments

- We explored several different network architectures for generating representations.

\[ \text{Denoted as } A \rightarrow B \]
Experiments
• We explored several different network architectures for generating representations

Denoted as $\text{Concat}(A,B) \rightarrow C$
Experiments

- We explored several different network architectures for generating representations

Denoted as $A \rightarrow \text{Concat}(B,C)$
Experiments

- We explored several different network architectures for generating representations

Denoted as

\[ \text{Concat}(A,B) \rightarrow \text{Concat}(B,C) \]
We explored several different network architectures for generating representations.

Denoted as: $\text{Embed}(A, B) \rightarrow C$
Experiments

- Unimodal Baseline

<table>
<thead>
<tr>
<th>Method</th>
<th>Feature</th>
<th>\begin{tabular}{c} BINARY ((-1, +1)) \end{tabular}</th>
<th>\begin{tabular}{c} 7-CLASS ((-3, \ldots, +3)) \end{tabular}</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Prec</td>
<td>Recall</td>
</tr>
<tr>
<td>UniModal-Baseline</td>
<td>Text (T)</td>
<td>0.77</td>
<td>0.76</td>
</tr>
<tr>
<td></td>
<td>Audio (A)</td>
<td>0.56</td>
<td>0.56</td>
</tr>
<tr>
<td></td>
<td>Video (V)</td>
<td>0.57</td>
<td>0.47</td>
</tr>
</tbody>
</table>

\[ T = \text{Text Modality, } A = \text{Audio Modality, } V = \text{Visual (facial) modality} \]
# Results (Bi-Modal)

- Bimodal Baseline & Experimental Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Feature</th>
<th>BINARY (-1, +1)</th>
<th>7-CLASS (-3, ..., +3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Prec Recall F1</td>
<td>Prec Recall F1</td>
</tr>
<tr>
<td>BiModal-Baseline</td>
<td>concat(T + V)</td>
<td>0.78 0.67 0.55</td>
<td>0.01 0.16 0.05</td>
</tr>
<tr>
<td></td>
<td>concat(T + A)</td>
<td>0.44 0.66 0.53</td>
<td>0.02 0.15 0.04</td>
</tr>
<tr>
<td></td>
<td>concat(A + V)</td>
<td>0.55 0.47 0.48</td>
<td>0.13 0.16 0.11</td>
</tr>
<tr>
<td></td>
<td>T → V</td>
<td>0.67 0.67 0.67</td>
<td>0.26 0.22 0.22</td>
</tr>
<tr>
<td></td>
<td>T → A</td>
<td>0.66 0.64 0.65</td>
<td>0.28 0.24 0.18</td>
</tr>
<tr>
<td></td>
<td>A → T</td>
<td>0.55 0.60 0.56</td>
<td>0.17 0.34 0.11</td>
</tr>
<tr>
<td></td>
<td>A → V</td>
<td>0.55 0.55 0.54</td>
<td>0.16 0.18 0.16</td>
</tr>
<tr>
<td></td>
<td>V → T</td>
<td>0.58 0.58 0.58</td>
<td>0.05 0.16 0.08</td>
</tr>
<tr>
<td></td>
<td>V → A</td>
<td>0.58 0.62 0.58</td>
<td>0.12 0.17 0.01</td>
</tr>
</tbody>
</table>

* T = Text Modality, A = Audio Modality, V = Visual (facial) modality
## Results (Bi-Modal)

- Bimodal Baseline & Experimental Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Feature</th>
<th>BINARY (−1, +1)</th>
<th>7-CLASS (−3, ..., +3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Prec</td>
<td>Recall</td>
</tr>
<tr>
<td>BiModal-Baseline</td>
<td>concat(T + V)</td>
<td>0.78</td>
<td>0.67</td>
</tr>
<tr>
<td></td>
<td>concat(T + A)</td>
<td>0.44</td>
<td>0.66</td>
</tr>
<tr>
<td></td>
<td>concat(A + V)</td>
<td>0.55</td>
<td>0.47</td>
</tr>
<tr>
<td>T → V</td>
<td></td>
<td>0.67</td>
<td>0.67</td>
</tr>
<tr>
<td>T → A</td>
<td></td>
<td>0.66</td>
<td>0.64</td>
</tr>
<tr>
<td>A → T</td>
<td></td>
<td>0.55</td>
<td>0.60</td>
</tr>
<tr>
<td>A → V</td>
<td></td>
<td>0.55</td>
<td>0.55</td>
</tr>
<tr>
<td>V → T</td>
<td></td>
<td>0.58</td>
<td>0.58</td>
</tr>
<tr>
<td>V → A</td>
<td></td>
<td>0.58</td>
<td>0.62</td>
</tr>
</tbody>
</table>

T = Text Modality, A = Audio Modality, V = Visual (facial) modality

10 Point Boost
# Results (Bi-Modal)

- Bimodal Baseline & Experimental Results

<table>
<thead>
<tr>
<th>Method</th>
<th>Feature</th>
<th>BINARY (−1, +1)</th>
<th>7-CLASS (−3, ..., +3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Prec Recall F1</td>
<td>Prec Recall F1</td>
</tr>
<tr>
<td>BiModal-Baseline</td>
<td><code>concat(T + V)</code></td>
<td>0.78 0.67 0.55</td>
<td>0.01 0.16 0.05</td>
</tr>
<tr>
<td></td>
<td><code>concat(T + A)</code></td>
<td>0.44 0.66 0.53</td>
<td>0.02 0.15 0.04</td>
</tr>
<tr>
<td></td>
<td><code>concat(A + V)</code></td>
<td>0.55 0.47 0.48</td>
<td>0.13 0.16 0.11</td>
</tr>
<tr>
<td></td>
<td>T → V</td>
<td>0.67 0.67 0.67</td>
<td>0.26 0.22 <strong>0.22</strong></td>
</tr>
<tr>
<td></td>
<td>T → A</td>
<td>0.66 0.64 0.65</td>
<td><strong>0.28</strong> 0.24 0.18</td>
</tr>
<tr>
<td>BiModal-Seq2Seq</td>
<td>A → T</td>
<td>0.55 0.60 0.56</td>
<td>0.17 <strong>0.34</strong> 0.11</td>
</tr>
<tr>
<td></td>
<td>A → V</td>
<td>0.55 0.55 0.54</td>
<td>0.16 0.18 0.16</td>
</tr>
<tr>
<td></td>
<td>V → T</td>
<td>0.58 0.58 0.58</td>
<td>0.05 0.16 0.08</td>
</tr>
<tr>
<td></td>
<td>V → A</td>
<td>0.58 0.62 0.58</td>
<td>0.12 0.17 0.01</td>
</tr>
</tbody>
</table>

12 Point Boost

*T = Text Modality, A = Audio Modality, V = Visual (facial) modality*
## Results (Tri-Modal)

- **Trimodal Baseline & Experimental Results**

<table>
<thead>
<tr>
<th>Method</th>
<th>Feature</th>
<th>BINARY (-1, +1)</th>
<th>7-CLASS (-3, ..., +3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Prec</td>
<td>Recall</td>
</tr>
<tr>
<td>TriModal-Baseline</td>
<td>concat(T + V + A)</td>
<td><strong>0.75</strong></td>
<td><strong>0.75</strong></td>
</tr>
<tr>
<td></td>
<td>embed(T, V) → A</td>
<td>0.56</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>embed(T, A) → V</td>
<td>0.60</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>embed(A, V) → T</td>
<td>0.66</td>
<td>0.53</td>
</tr>
<tr>
<td></td>
<td>embed(A, T) → V</td>
<td>0.59</td>
<td>0.51</td>
</tr>
<tr>
<td></td>
<td>embed(V, T) → A</td>
<td>0.59</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>embed(V, A) → T</td>
<td>0.57</td>
<td>0.61</td>
</tr>
<tr>
<td>TriModal-Seq2Seq</td>
<td>concat(T, V) → A</td>
<td>0.67</td>
<td>0.66</td>
</tr>
<tr>
<td></td>
<td>concat(A, T) → V</td>
<td>0.54</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>concat(V, A) → T</td>
<td>0.59</td>
<td>0.59</td>
</tr>
<tr>
<td></td>
<td>T → concat(A, V)</td>
<td>0.70</td>
<td>0.65</td>
</tr>
<tr>
<td></td>
<td>A → concat(T, V)</td>
<td>0.55</td>
<td>0.53</td>
</tr>
<tr>
<td></td>
<td>concat(T, A) → concat(T, V)</td>
<td>0.62</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>concat(T, V) → concat(T, A)</td>
<td>0.68</td>
<td>0.70</td>
</tr>
</tbody>
</table>

*T = Text Modality, A = Audio Modality, V = Visual (facial) modality*
Results - Takeaways

- We clearly outperform the baselines in the **bi-modal** domain
  - In the 7-class paradigm we often outperform by a large margin
  - For datasets without transcripts this approach may result in significant gains

- Slightly outperform baseline in tri-modal multiclass setting

- Significantly longer training times than the baseline alone
Future Work

- Our method is unsupervised, we will pre-train seq2seq model with external dataset
- Use variational seq2seq to refine the training
- Further explore end-to-end training
- Explore additional methods for encoding sequences with 2 modalities
  - Multi-view LSTM
Acknowledgements

- Amir Zadeh
- Volkan Cirik
- Louis-Philippe Morency
- Somya Wadhwa
- Minghai Chen
- Hieu Pham
- Workshop Reviewers
- *AWS*

Thank you!
Appendix
Problem Formulation

- Input: $X = (X_1, X_2, \ldots, X_n)$ where $X_i = (X_i^{text}, X_i^{audio}, X_i^{video})$
- Output: $Y = (Y_1, Y_2, \ldots, Y_n)$, $Y_i \in \mathbb{R}$
- Align based on word-level
  \[
  X_i^{text} = (w_i^{(1)}, w_i^{(2)}, \ldots, w_i^{(T_i)})
  \]
  \[
  X_i^{audio} = (a_i^{(1)}, a_i^{(2)}, \ldots, a_i^{(T_i)})
  \]
  \[
  X_i^{video} = (v_i^{(1)}, v_i^{(2)}, \ldots, v_i^{(T_i)})
  \]
- Goal: Learn the embedding representation
  \[
  \tilde{X}_i = f(X_i) = f((X_i^{text}, X_i^{audio}, X_i^{video}))
  \]
  \[
  \tilde{X}_i = f(X_i) = Seq2Seq\_Encoder(X_i)
  \]
Problem Formulation (cont’d)

- Transformed input: \( \tilde{X} = (\tilde{x}^1, \tilde{x}^2, ..., \tilde{x}^T) \) with output \( Y = (y^1, y^2, ..., y^T) \)
- Using RNN with K hidden layers:
  \[
  h = (h^1, h^2, \ldots, h^K)
  \]
  \[
  h^k = (h^k_1, h^k_2, \ldots, h^k_D), \ k \in [1, K]
  \]
- First layer: \( h^1_t = H(W_{xh^1} \tilde{x}_t + W_{h^1h^1} h^1_{t-1} + b_{h^1}) \)
- Layer k: \( h^k_t = H(W_{h^{k-1}h^k} h^{k-1}_t + W_{h^kh^k} h^k_{t-1} + b_{h^k}) \)
- Using soft attention at last hidden layer K:
  \[
  \alpha = \text{softmax} \left( \begin{bmatrix}
  W_\alpha h^K_1 \\
  W_\alpha h^K_2 \\
  \vdots \\
  W_\alpha h^K_T
\end{bmatrix} \right)
  \]
Problem Formulation (cont’d)

- Output of last hidden layer: \(A = [h^K_1, h^K_2, \ldots, h^K_T] \alpha = H^K \alpha\)

- Final output: \(\tilde{y}_t = W_{Ay} A + b_y\)

- Mean Absolute Error Loss: \(\mathbb{L}_{MAE}(\tilde{Y}, Y) = \mathbb{E}[|\tilde{Y} - Y|]\)

- Model is trained with SGD
Algorithm 1 Seq2Seq Modality Translation

$X, Y, S$ are 2 modalities and sentiment sequences

1: Phase 1: Train Seq2Seq
2: $\mathcal{E}_{XY} \leftarrow \text{Seq2Seq.RNN.Encode}(X)$
3: $\tilde{Y} \leftarrow \text{Seq2Seq.RNN.Decode}(\mathcal{E}_{XY})$
4: $\text{loss} = \text{cross_entropy}(\tilde{Y}, Y)$
5: Backprop to update params

6: Phase 2: Sentiment Regression
7: $\mathcal{E}_{XY} \leftarrow \text{Seq2Seq.RNN.Encode}(X)$ ▶ trained encoder in Seq2Seq model
8: $R = \text{RNN}(\mathcal{E}_{XY})$
9: $\text{score} \leftarrow \text{Regression}(R)$
10: $\text{loss} \leftarrow \text{MAE}(\text{score}, S)$
11: Backprop to update params
Hierarchical Seq2Seq Modality Translation

Algorithm 2 Hierarchical Seq2Seq Modality Translation: $X, Y, Z, S$ are 3 modalities and sentiment sequences

1: Phase 1: Train Seq2Seq for 2 modalities
2: $\mathcal{E}_{XY} \leftarrow$ Seq2Seq_RNN_Encode($X$)
3: $\tilde{Y} \leftarrow$ Seq2Seq_RNN_Decode($\mathcal{E}_{XY}$)
4: $\text{loss} = \text{cross_entropy}(\tilde{Y}, Y)$
5: Backpropagate to update parameters

6: Phase 2: Train Seq2Seq for 3 modalities
7: $\mathcal{E}_{XYZ} \leftarrow$ Seq2Seq_RNN_Encode($\mathcal{E}_{XY}$)
8: $\tilde{Z} \leftarrow$ Seq2Seq_RNN_Decode($\mathcal{E}_{XYZ}$)
9: $\text{loss} = \text{cross_entropy}(\tilde{Z}, Z)$
10: Backpropagate to update parameters

11: Phase 3: Sentiment Regression
12: $\mathcal{E}_{XYZ} \leftarrow$ Seq2Seq_RNN_Encode($\mathcal{E}_{XY}$)
13: $R = \text{RNN}(\mathcal{E}_{XYZ})$
14: $\text{score} \leftarrow \text{Regression}(R)$
15: $\text{loss} \leftarrow \text{MAE}(\text{score}, S)$
16: Backpropagate to update parameters