DeepTennis: Mid-Match Tennis Predictions

Dipika Badri,[1] Sven Lerner,[1] Kevin Monogue[1,2]

[1]Electrical Engineering, Stanford University
[2,3] ICME, Stanford University

Stanford
CS 230

Motivation / Summary

We built a recurrent neural network utilizing the Long Short-Term Memory (LSTM) model to compute 'live' win probabilities for tennis matches mid-game. Using a detailed point-by-point dataset [1] for the four Grand Slam tennis tournaments, our model is designed to learn deep relationships behind the sequential data and to predict the probability of winning the match for each player after every point.

Data

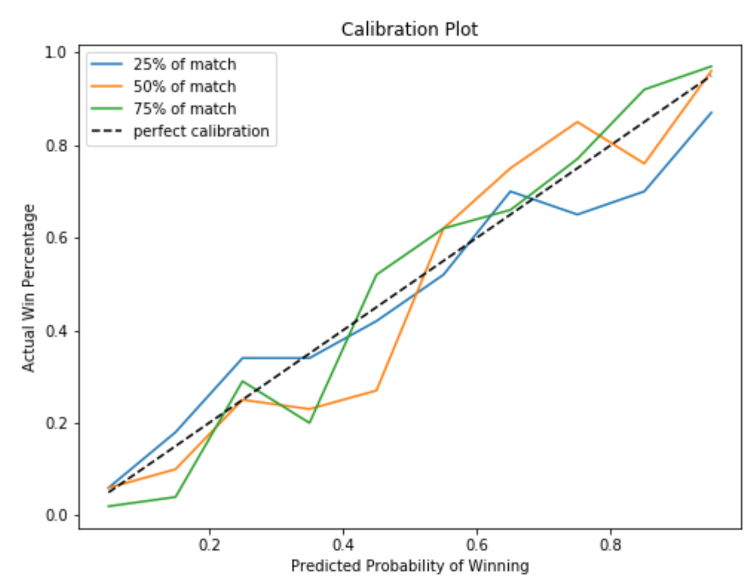
```
match_id                  2019-usopen-1101
player1                     Novak Djokovic
player2            Roberto Carballes Baena
winner                                   1
SetNo                                    2
P2Winner                                 0
P1DoubleFault                            0
P2DoubleFault                            0
P1UnfErr                                 0
P2UnfErr                                 0
P1NetPoint                               0
P2NetPoint                               0
P1NetPointWon                            0
P2NetPointWon                            0
P1BreakPoint                             1
P2BreakPoint                             0
P1BreakPointWon                          1
P2BreakPointWon                          0
Speed_MPH                              116
RallyCount                               4
P1DistanceRun                       14.894
P2DistanceRun                       17.513
p1_sets_to_win                           2
p2_sets_to_win                           3
p1_games_to_win                          7
p2_games_to_win                         17
```

We used the point-by-point dataset for the tennis Grand Slams (Wimbledon, US Open, French Open, and Australian Open) provided by Jeff Sackman [1] along with derived data such as minimum sets to win. We also incorporated pre-match win probabilities for each player from a paper on pre-match tennis win predictions (Gollub).

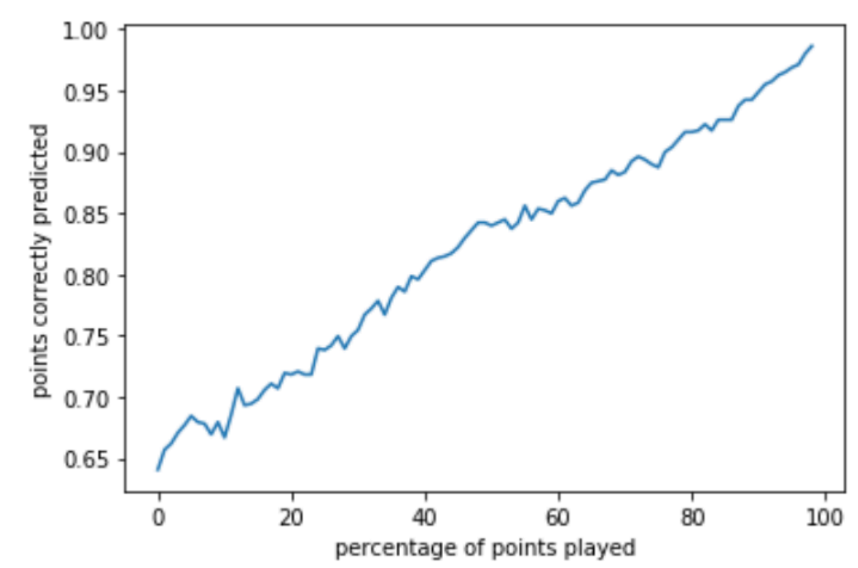
Results and Experiments

x	DeepTennis	K-M Logit Elo (Gollub)	Logistic Regression
Model Accuracy (points correctly predicted)	79.5%	76.5%	73.4%

Model Calibration

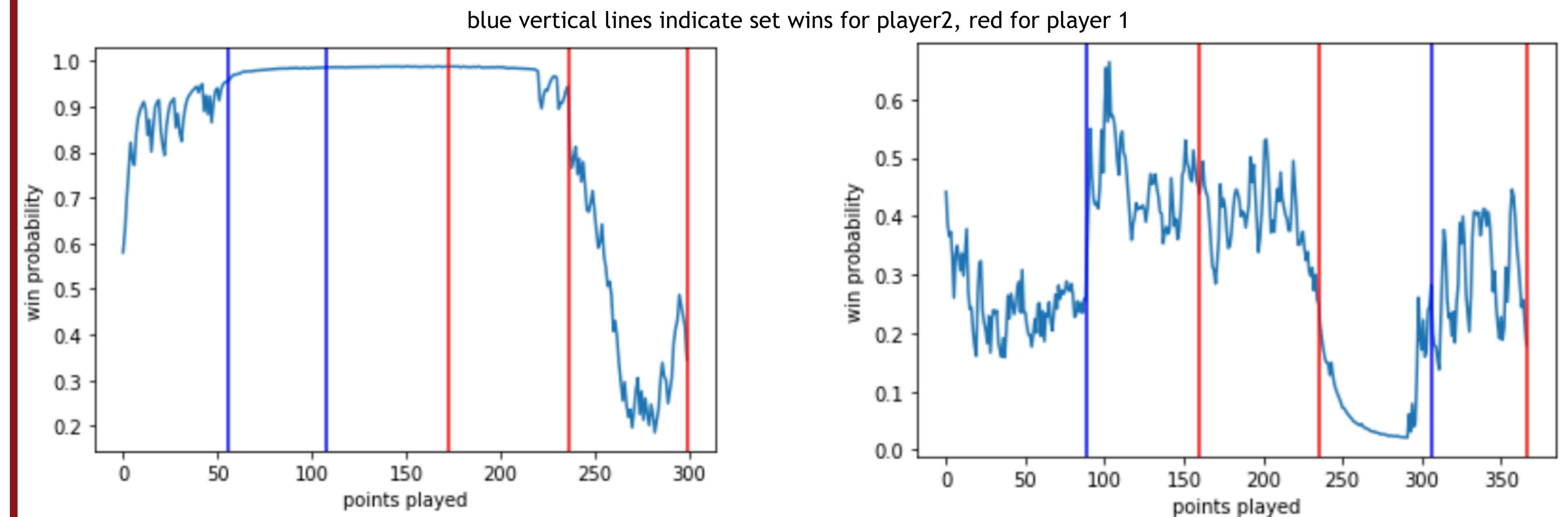


Model Accuracy over Match



Thiem vs. Gulbis, US Open 2014

Djokovic vs. Federer, Wimbledon 2014



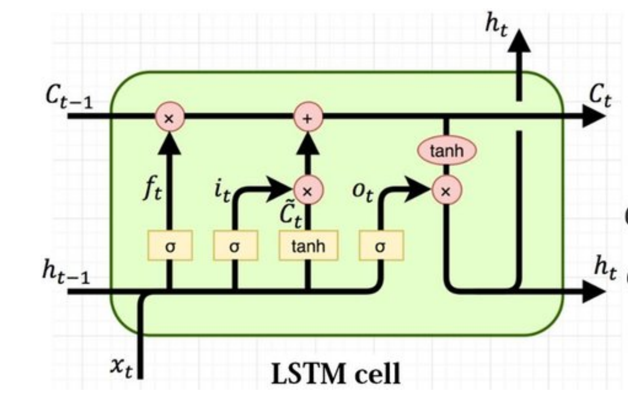
Our model assigned high win probability for Gulbis for the majority of the match - he was a heavy favorite over Thiem, who was participating in his first ever Grand Slam. Thiem only became favored when being mere points away from pulling off the upset.

The 2014 Wimbledon final exhibits our models ability to detect momentum well, especially when the match involves similar level players. Federer first set win was a crucial head start, before Djokovic nearly put the match away until a back-and-forth final set.

x	Prematch	After Set 1	After Set 2	After Set 3	After Set 4
DeepTennis Prediction Accuracy	76%	84%	85%	93%	85%

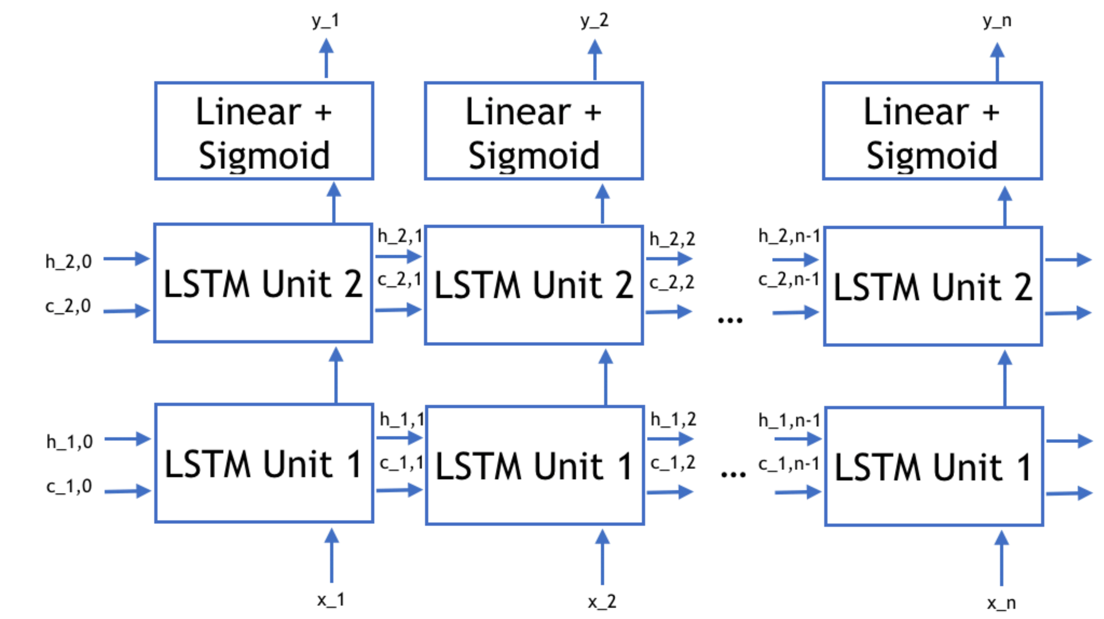
By definition, after set 4, the match is tied and has become a best of one series. Our model exhibits a 9 percent accuracy boost over the prematch win probability when predicting this decisive set, illustrating that it has learned important features over the course of the match.

Model Architecture



$$i_t = \sigma(x_t U^i + h_{t-1} W^i)$$
$$f_t = \sigma(x_t U^f + h_{t-1} W^f)$$
$$o_t = \sigma(x_t U^o + h_{t-1} W^o)$$
$$\tilde{C}_t = \tanh\left(x_t U^g + h_{t-1} W^g\right)$$
$$C_t = \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t)$$
$$h_t = \tanh(C_t) * o_t$$

[3]



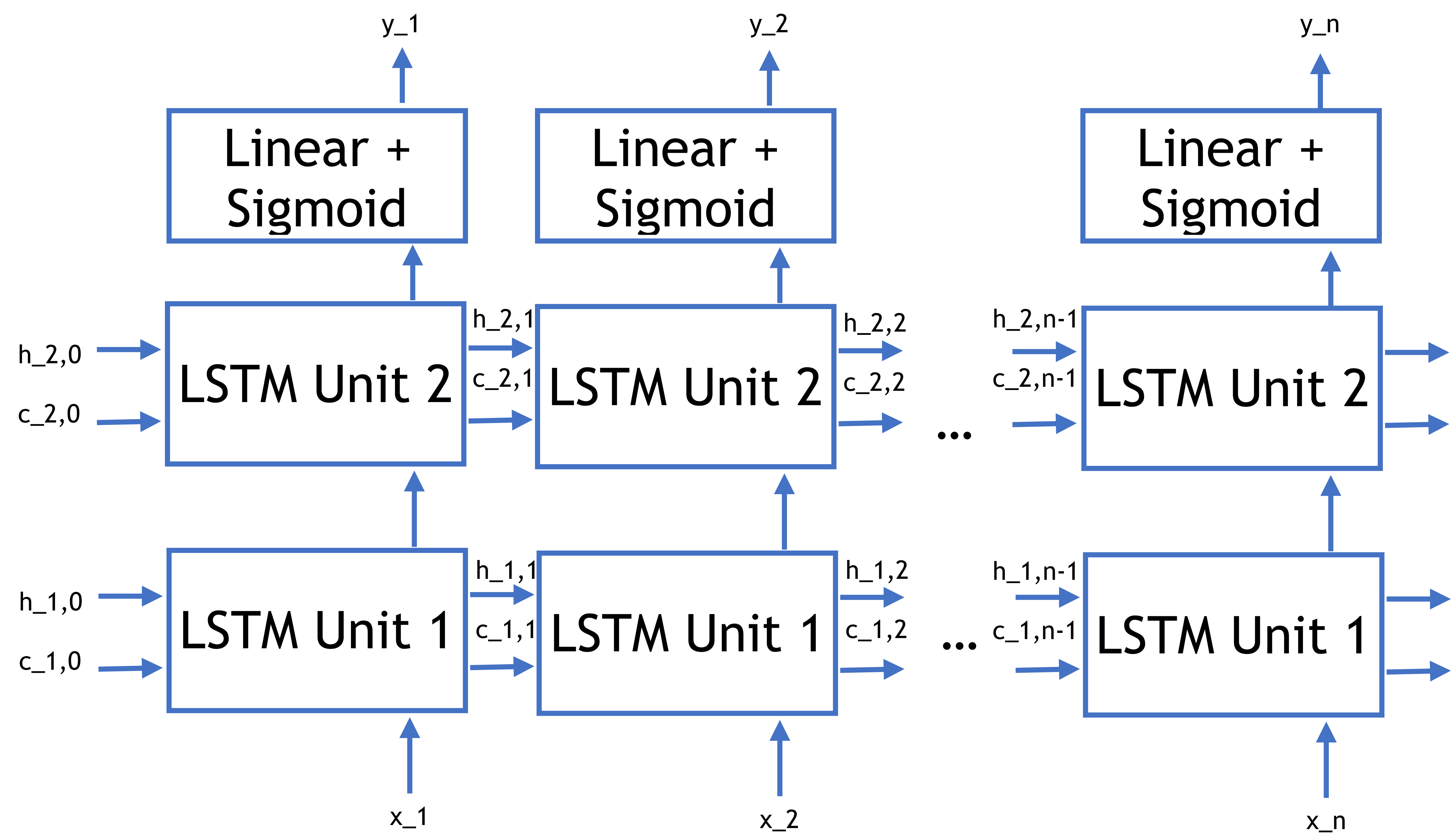
We utilized an architecture consisting of stacked LSTM units follows by a single linear output layer with sigmoid activation.

Conclusion / Future Directions

The implementation of a two-layer, 50 hidden node LSTM model appears to have improved performance over existing methodologies for mid-match tennis prediction. We believe our model's performance could be further improved with the incorporation of more data or more detailed features, and future work should involve the collection of that data.

References

[1] Courtesy Jeff Sackman: https://github.com/JeffSackmann/tennis_slam_pointbypoint, [2] Gollub, Jacob. Producing Win Probabilities for Professional Tennis Matches from any Score. Diss. 2019. [3]Varsamopoulos, Savvas, Koen Bertels, and Carmen G. Almudever. "Designing neural network based decoders for surface codes." arXiv preprint arXiv: 1811.12456 (2018).



y_1
y_2
y_n
Linear + Sigmoid
Linear + Sigmoid
Linear + Sigmoid
h_2,0
c_2,0
LSTM Unit 2
h_2,1
c_2,1
LSTM Unit 2
h_2,2
c_2,2
...
h_2,n-1
c_2,n-1
LSTM Unit 2
h_1,0
c_1,0
LSTM Unit 1
h_1,1
c_1,1
LSTM Unit 1
h_1,2
c_1,2
...
h_1,n-1
c_1,n-1
LSTM Unit 1
x_1
x_2
x_n