**RNN headline generator** **Aaron Liu, 2019/04/07**

# Objective:

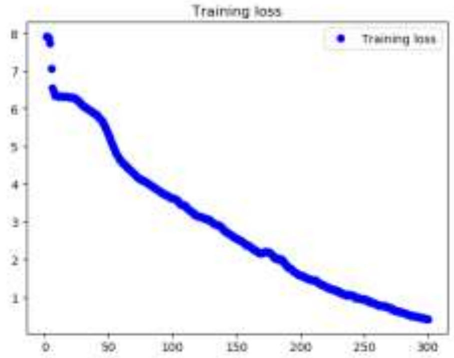
I am trying to do multi-classification of the data of news headlines published over 15 years by using multiple RNN Classifiers and make efforts to experiment with preprocessing the input data and network structure.

In the python file, first part is the libraries I used for my multi-classification modeling, then is the data-loaded part and the data-pretreated part. After these three parts, I would use multiple RNN Classifiers and experiment with various parameters and input to explore and do multi-classification.

# Generate text by word with original model

**Training Procedure:**
My original RNN model has four hidden layers, including one embedding layer because of word-level representation, and one fully connected layer. Since I need to predict text by size of vocabulary, I use 'softmax' as the activation function for the output layer to do multi-classification. For the training algorithm, I use "adam" as the optimizer and "categorical_crossentropy" as the loss function. I also chose to put "accuracy" in the metrics.

**The following chart show the training loss along with training iterations:**



**Produce headlines of 4 words for articles:**

1. consent disbanded illegal fees
2. death cash says expansion
3. strike charges for laws
4. life budget future poll
5. supply and support Somalis
6. new children woman cars
7. lump media talks crash
8. study supply injury teen
9. outbreak blitz for children

**Produce headlines of 5 words for articles:**

10. aside mcmurdo cause Tasmania burning
11. woman support Howard activists terrorism

12. embassy more shooting shifted Australia

**Produce headlines of 7 words for articles:**

13. burning stocks Australians announced xi Tasmania burning
14. say more listing west budget valley one
15. Korea plans for children west police west

**The best headline:**

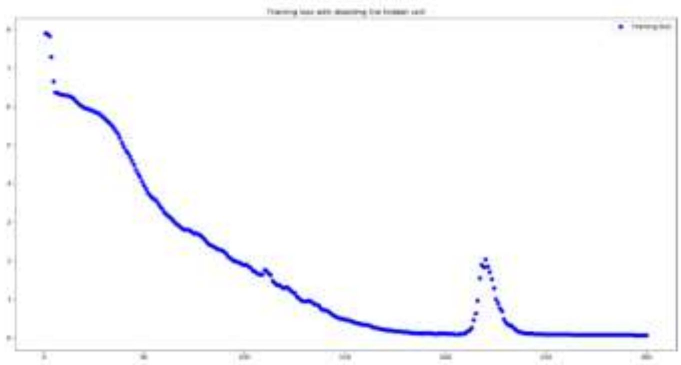
Korea plans for children west police west

# Generate text by word with model doubling the number of hidden units

**Training Procedure:**

My RNN model which doubles the number of hidden units has four hidden layers, including one embedding layer because of word-level representation, and one fully connected layer. The only difference of this model is that I double the number of hidden units to become 200 this time.

**The following chart show the training loss along with training iterations:**



**The change of loss shows how well my RNN learns through time:**

Epoch 00060: loss improved from 3.85362 to 3.76894

Epoch 00120: loss did not improve from 1.46132

Epoch 00180: loss improved from 0.05606 to 0.05448

Epoch 00240: loss improved from 0.01669 to 0.01645

Epoch 00300: loss improved from 0.00796 to 0.00787

**Produce headlines of 4 words for articles:**

1. workers Sydney tour system
2. debut campaign crash rules
3. audit alert friendly service
4. judge agreements over crash
5. north process overdose education
6. development citizenship is crash
7. over danger drinking transplant
8. fraud day assault cases
9. rainforest money delegates disbanded
10. elections change over sites
11. told support tour rules
12. yoga lifetime front spirituality

**Produce headlines of 7 words for articles:**

13. explosion lawyer’s hospital health over coastal waterfront
14. election over mystery incident soldier’s marathon experience

**The best headline:**

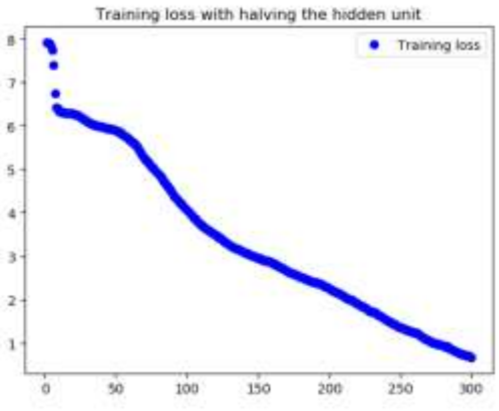
elections change over sites

# Generate text by word with model halving the number of hidden units

**Training Procedure:**

My RNN model which halves the number of hidden units has four hidden layers, including one embedding layer because of word-level representation, and one fully connected layer. The only difference of this model is that I halve the number of hidden units to become only 50 this time.

**The following chart show the training loss along with training iterations:**



**The change of loss shows how well my RNN learns through time:**

Epoch 00060: loss improved from 5.47420 to 5.43936

Epoch 00120: loss improved from 3.52163 to 3.50061

Epoch 00180: loss improved from 1.37473 to 1.35464

Epoch 00240: loss improved from 0.51513 to 0.50423

Epoch 00300: loss improved from 0.14522 to 0.14249

**Produce headlines of 4 words for articles:**

1. test of restaurant rescue
2. victim allegations Nemtsov recent
3. judges hopes evidence heard
4. restaurant protest scam tools
5. opening comments inland tallies
6. exports win hospitals care
7. owed hospital qld truck
8. out years on weather
9. spot ABC close causes
10. are their plans inad
11. well prisoners suffering family
12. defence identified continues protest

**Produce headlines of 7 words for articles:**

13. we care hospitals care qld leg Bangkok
14. friends concert owed test to school restaurant

**The best headline:**

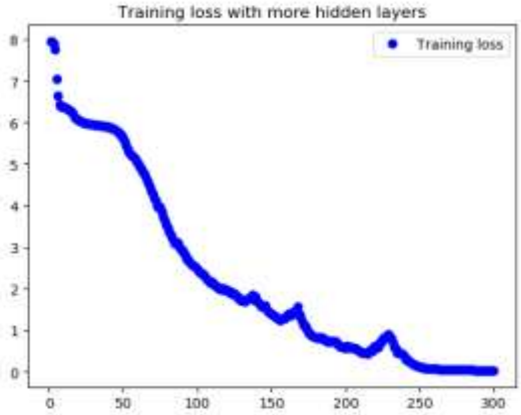
judges hopes evidence heard

# Generate text by word with model adding one more hidden layer

**Training Procedure:**

My RNN model which doubles the number of hidden units has four hidden layers, including one embedding layer because of word-level representation, and one fully connected layer. The only difference of this model is that I double the number of hidden units to become 200 this time.

**The following chart show the training loss along with training iterations:**



**The change of loss shows how well my RNN learns through time:**

Epoch 00060: loss improved from 4.67782 to 4.66460

Epoch 00120: loss did not improve from 2.02224

Epoch 00180: loss improved from 0.37442 to 0.35049

Epoch 00240: loss improved from 0.02053 to 0.02018

Epoch 00300: loss improved from 0.00780 to 0.00769

**Produce headlines of 4 words for articles:**

1. funding sea patel arguments
2. safety auction tigers' prices
3. workers protest Zealand's loss
4. change for Queensland budgets
5. young nurse Chinese witnesses
6. protestors bulls revenue service
7. men bullying hospital resources
8. workers protest frog watch upgrade
9. fast origin study services
10. ABC proposed origin poll
11. impact cbd Egypt weather

**Produce headlines of 7 words for articles:**

12. apartments of suit win owners breathing service
13. boost centre nurse destroyed Tasmania fish crackdown

**The best headline:**

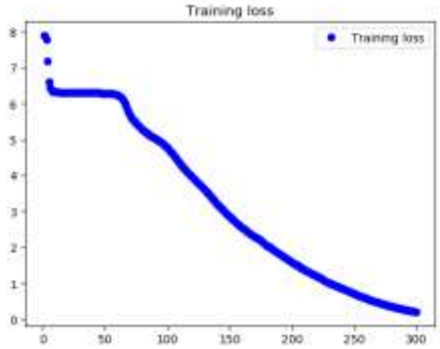
men bullying hospital resources

## Generate text by word with model doubling the length of sequence

**Training Procedure:**

My RNN model which doubles the length of sequence has four hidden layers, including one embedding layer because of word-level representation, and one fully connected layer. The only difference of this model is that I double the length of sequence to become 16 this time.

**The following chart show the training loss along with training iterations:**



**The change of loss shows how well my RNN learns through time:**

Epoch 00060: loss improved from 6.13438 to 6.09171

Epoch 00120: loss improved from 3.65025 to 3.63319

Epoch 00180: loss improved from 1.09326 to 1.06702

Epoch 00240: loss improved from 0.15665 to 0.15202

Epoch 00300: loss improved from 0.04454 to 0.04376

**Produce headlines of 4 words for articles:**

1. murder move to not
2. cull the south fight
3. property soldiers spurs marine
4. economy bay policy sentence
5. surprise dad Florida beach
6. robberies Indian murder time
7. is Australia weekend leaving
8. tensions costs India says
9. Indian murder time politician

**Produce headlines of 5 words for articles:**

10. attack back perth flow campaign
11. savings centres this weekend fwa
12. ban property refuge spurs marine

**Produce headlines of 7 words for articles:**

13. crash drones new savings centres this weekend
14. to homeless Michelle India of the fight
15. arsonist stop removed prizes tortured way council

**The best headline:**

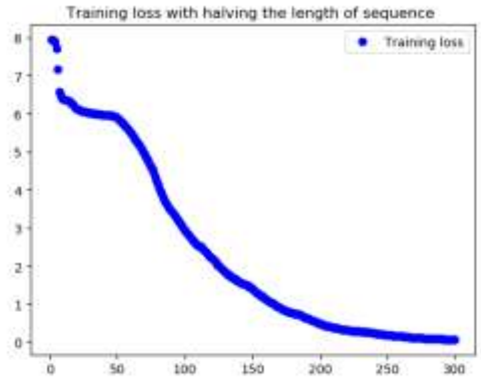
Is Australia weekend leaving?

# Generate text by word with model halving the length of sequence

**Training Procedure:**

My RNN model which halves the length of sequence has four hidden layers, including one embedding layer because of word-level representation, and one fully connected layer. The only difference of this model is that I halve the length of sequence to become only 4 this time.

**The following chart show the training loss along with training iterations:**



**The change of loss show how well my RNN learns through time:**

Epoch 00060: loss improved from 5.30353 to 5.24166

Epoch 00120: loss improved from 2.62832 to 2.59248

Epoch 00180: loss improved from 0.26076 to 0.25387

Epoch 00240: loss improved from 0.03664 to 0.03617

Epoch 00300: loss improved from 0.01302 to 0.01283

**Produce headlines of 4 words for articles:**

1. card stats billboard giants
2. developments inquiry attacks scandal
3. approved Ipswich billboard performers
4. safely police papers inquiry
5. boosting new million trees
6. developments diggers the report
7. hearing doctors signing recording
8. cuts funding report coast
9. future dyball moved reality
10. residents approved billboard trump
11. boost from earthquake death

**Produce headlines of 7 words for articles:**

12. connections people bombers bullying blacks grid calls
13. attacks land worst vexatious police handbags buyers
14. Singapore doubt numbers market era public recording
15. calls Singapore search escape unfit police livestock

**The best headline:**

boosting new million trees

## Conclusion:

**The funniest headline overall:** men bullying hospital resources
**The best of the best headline:** elections change over sites

After experimenting with number of hidden units, number of hidden layers and length of sequence, I found that it takes more time to run my RNN model if I increased the number of input or added more hidden layers. Also, I found out that I could generate better headlines when I increased the number of hidden units, hidden layers or length of sequence. For example, "Elections change over sites" and "Is Australia weekend leaving?"**.**

Next time, I might try to train and experiment my RNN model with different numbers of units, different length of sequence, adding more hidden layers and dropping out some features although it would take more time. Also, I would do cross validation to find the optimal epoch, batch size and parameters to see if I could generate better and more logical headlines.

## Appendix:

- Deep Learning with Python, Francois Cholet
- A Million News Headlines: News headlines published over a period of 15 years. **https://www.kaggle.com/therohk/million-headlines**