HierNet, MILNET and Variants

towards a better vectorial representation of documents

Louis (Yiqing) Luo

July 31st, 2018

Motivation

Objective: To construct vectors that can capture rich sentiment features for documents (a collection of sentences)

- Sentiment classification is a very popular and powerful problem in Natural Language Processing
- Vector representation of documents can potentially be used for further research

Motivation

Objective: To construct vectors that can capture rich sentiment features for documents (a collection of sentences)

- Sentiment classification is a very popular and powerful problem in Natural Language Processing
- Vector representation of documents can potentially be used for further research

Sentiment Score: 0

If, as promised, this movie was restored then the results are simply horrible. Instead of a intelligent restoration what this people did was to tint every scene to red, probably with photoshop, with disastrous results. Douglas m. music is as unremarkable as the bad restoration. His accompaniment does not enhance the images at all. In all, the available print shown on tcm is unwatchable and i had to turn off the tv set.

Motivation

Objective: To construct vectors that can capture rich sentiment features for documents (a collection of sentences)

- Sentiment classification is a very popular and powerful problem in Natural Language Processing
- Vector representation of documents can potentially be used for further research

Contributions:

1. Publicly available code of the Multi-Instance Network (MILNET) in Keras (original code not publicly available and code from other resources does not exist)
2. Introduction of a variant of MILNET which yields vectors that are document-level representations
3. Application and analysis of state-of-the-art manifold learning techniques on embedding space coupled with state-of-the-art sentiment classifiers

Current Metrics

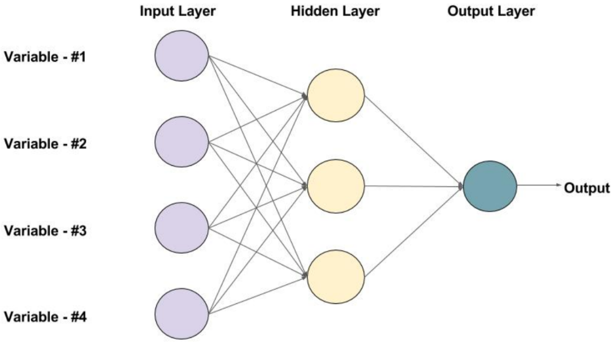
- Binary Classification on IMDB (Movie review) datasets
  - "1" - positive comment
  - "0" - negative comment
- A good model and a feature-rich document vector should both entail very good results for this test.

Outline

1. Tools in Neural Network
2. State-of-the-Art Models for Sentiment Analysis
3. Experiment Results I
4. Adding Manifold Learning to Embedding Space
5. Experiment Results II
6. Future Work - for next 2 weeks

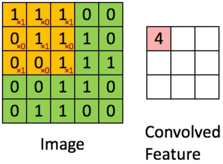
Brief Overview of Tools used in Neural Networks

1. Feed-Forward Neural Network (FFNN)
   - $y = \sigma(\Sigma_i W_i * x_i + b_i)$
   - A collection of single neurons, where each neural is composed of a linear weight and bias, followed by an activation function.



Brief Overview of Tools used in Neural Networks

2. Convolution Layer (CNN)
   - $v_i = \sum_{n=0}^{N-1} f(n)g(x-n)$
   - each entry in output of CNN = output of a neuron that looks at only a small region in the input and shares parameters with all neurons to the left and right spatially



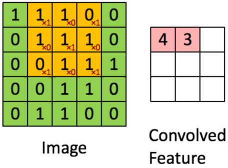
Image

$1_{\times1}$	$1_{\times0}$	$1_{\times1}$	0	0
$0_{\times0}$	$1_{\times1}$	$1_{\times0}$	1	0
$0_{\times1}$	$0_{\times0}$	$1_{\times1}$	1	1
0	0	1	1	0
0	1	1	0	0

Convolved Feature

4		

(a) $4*1+5*0=4$



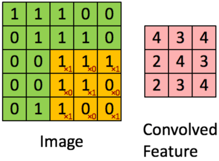
Image

1	$1_{\times1}$	$1_{\times0}$	$0_{\times1}$	0
0	$1_{\times0}$	$1_{\times1}$	$1_{\times0}$	0
0	$0_{\times1}$	$1_{\times0}$	$1_{\times1}$	1
0	0	1	1	0
0	1	1	0	0

Convolved Feature

4	3	

(b) $3*1+6*0=4$



Image

1	1	1	0	0
0	1	1	1	0
0	0	$1_{\times1}$	$1_{\times0}$	$1_{\times1}$
0	0	$1_{\times0}$	$1_{\times1}$	$0_{\times0}$
0	1	$1_{\times1}$	$0_{\times0}$	$0_{\times1}$

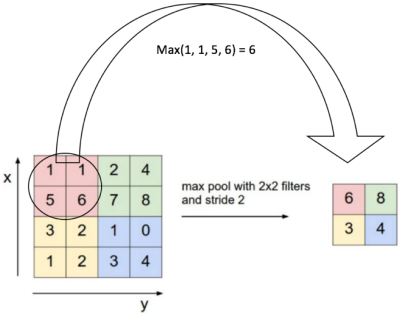
Convolved Feature

4	3	4
2	4	3
2	3	4

(c) $4*1+5*0=4$

Brief Overview of Tools used in Neural Networks

3. Max pooling Layer
   - $m_i = \max\limits_{j \epsilon neighbourhood(i)} v_j$
   - reduces the spatial size of the representation to reduce the amount of parameters and computation in the network

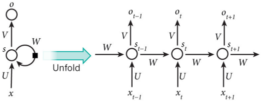


1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

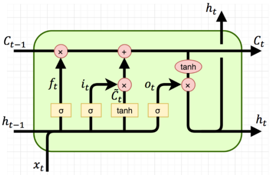
6	8
3	4

Brief Overview of Tools used in Neural Networks

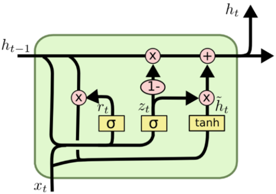
4. Recurrent Neural Network
   - $h_t = \sigma(W * h_{t-1} + U * x_t) + bias$
   - more generally: $h_t = f(h_{t-1}, x_t)$
   - Popular choice of function f: LSTM and GRU
   - uses their internal state (memory) to process sequences of inputs



(d) simple RNN



(e) Long-Short Term Memory (LSTM)



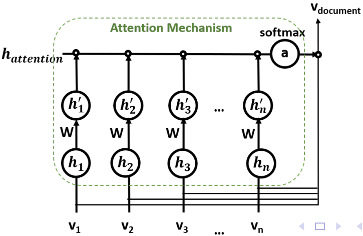
(f) Gated Recurrent Units (GRU)

Brief Review of Tools used in Neural Networks

5. Attention Weighting

$$\begin{cases} h_i' = tanh(W_{attention} * h_i + b_{attention}) \\ a_i = softmax(h_i'^{\mathbf{T}} h_{attention}) = \dfrac{e^{h_i'^{\mathbf{T}} h_{attention}}}{\Sigma_j e^{h_j'^{\mathbf{T}} h_{attention}}} \\ v_{document} = \Sigma_i a_i * v_i \end{cases}$$

- v_i is the input vector, and h_i is the vector from which the similarity score is computed
- Weights each input vector by the similarity score between the input and a predefined and trainable vector



Outline

1 Tools in Neural Network

2 State-of-the-Art Models for Sentiment Analysis

3 Experiment Results I

4 Adding Manifold Learning to Embedding Space

5 Experiment Results II

6 Future Work - for next 2 weeks

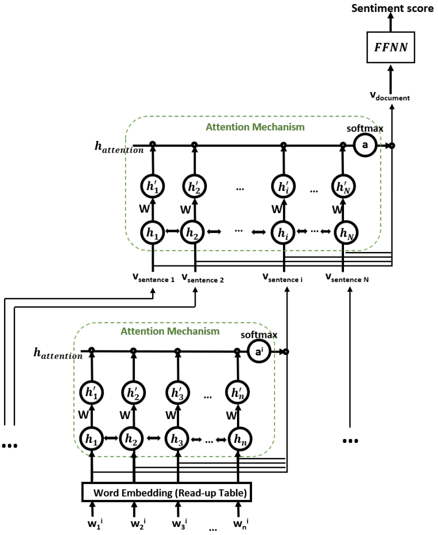
Current State-of-the-Art Models

1. Hierachy Network (HierNET) [Zichao Yang, Diyi Yang, Chris Dyer et al. 2016]
2. Multiple Instance Learning Network (MILNET) [Stefanos Angelidis and Mirella Lapata. 2018]

Hierachy Attention Network (HierNet)

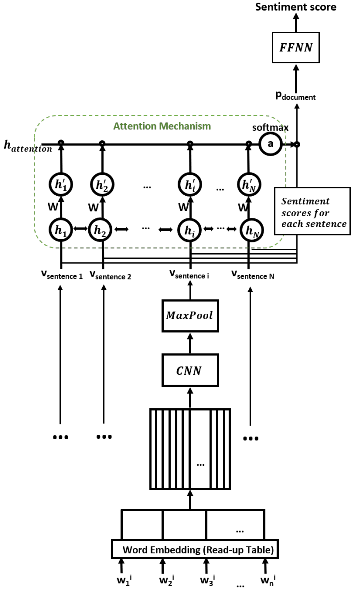
Hierachy - from lower-level structures to higher-level structures

1. Examines inter-relationships between words using Bi-RNN
2. Uses word-level attention to find important words and form sentence vector
3. Examines inter-relationships between sentences using Bi-RNN
4. Uses word-level attention to find important sentences and form document vector

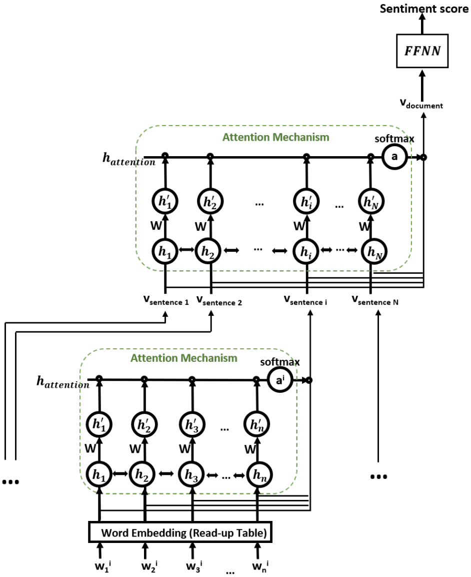


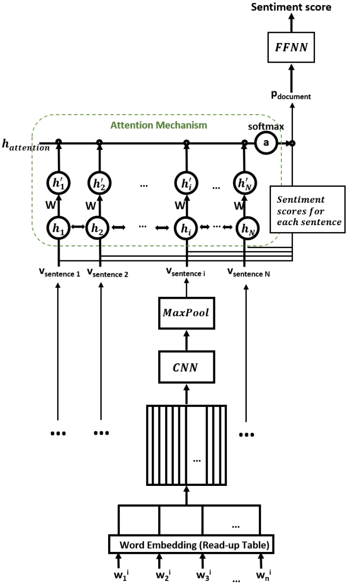
Multi-Instance Network (MILNET)

1. Examines relationship between nearby words within the same sentence in high-dimensional space using CNN
2. Predicts sentence-level sentiment using Artificial Neural Network
3. Uses sentence-level attention to find important sentences
4. Weighs all sentence-level sentiment using the attention weights



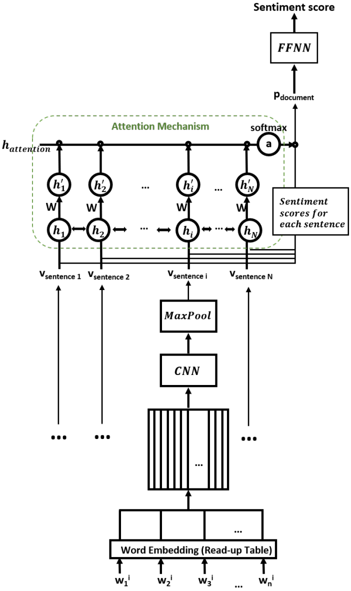
Left - HierNet; Right - MILNET





Comments about MILNET

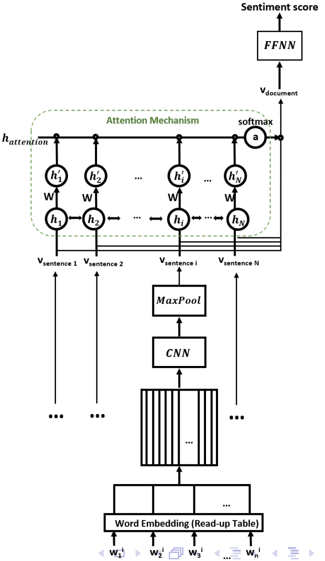
- No available code online.
- Does not produce a vectorial representation of the entire documents
- However, does produce vectorial representation of each sentences
- MILNET has the **state-of-the-art accuracy** for IMBD sentiment prediction



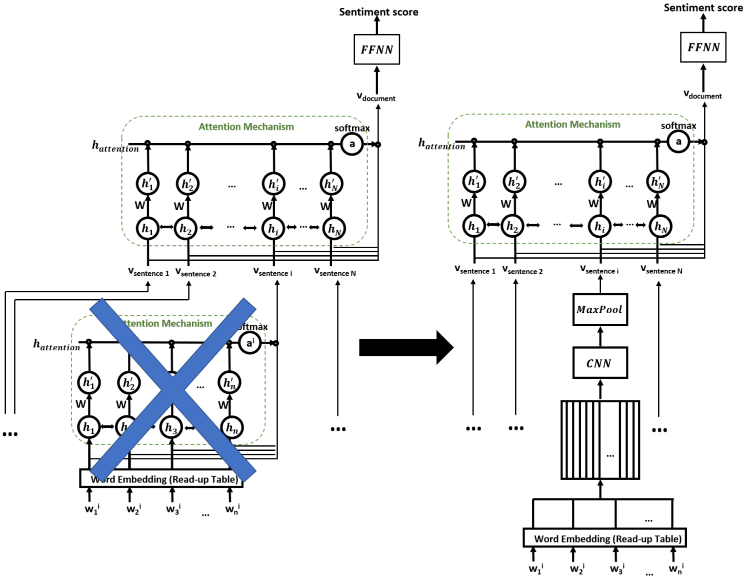
A Simple Variant of MILNET for Vectorial Representation of Document

Combines aspects of both HierNet and MILNET:

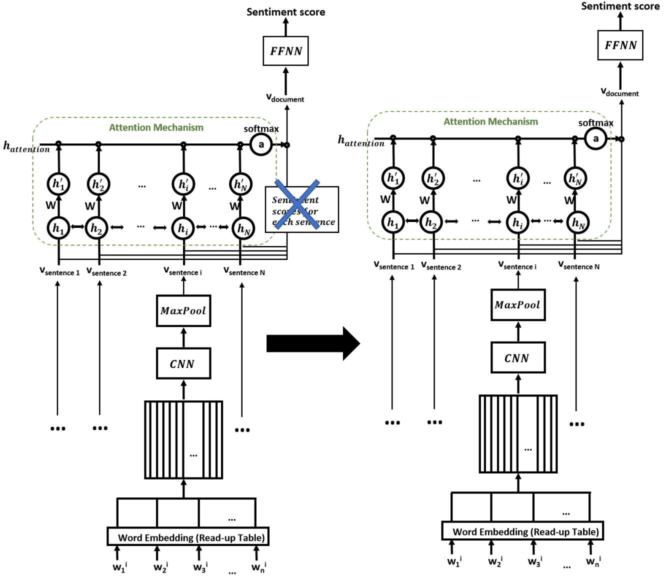
1. Uses CNN to examine relationships between words in each sentence [MILNET]
2. Examines inter-relationships between sentences using Bi-RNN [HIERNET]
3. Uses sentence-level attention to determine a document vector [HIERNET]



Comparison between HierNET and Variant



Comparison between MILNET and Variant


Sentiment score
FFNN
$v_{document}$
Attention Mechanism
softmax
a
$h_{attention}$
h'_1 h'_2 ... h'_i ... h'_N
W
h_1 h_2 ... h_i ... h_N
~~Sentiment scores for each sentence~~
$v_{sentence\ 1}$ $v_{sentence\ 2}$ $v_{sentence\ i}$ $v_{sentence\ N}$
MaxPool
CNN
...
Word Embedding (Read-up Table)
w_1^i w_2^i w_3^i ... w_n^i
Sentiment score
FFNN
$v_{document}$
Attention Mechanism
softmax
a
$h_{attention}$
h'_1 h'_2 ... h'_i ... h'_N
W
h_1 h_2 ... h_i ... h_N
$v_{sentence\ 1}$ $v_{sentence\ 2}$ $v_{sentence\ i}$ $v_{sentence\ N}$
MaxPool
CNN
...
Word Embedding (Read-up Table)
w_1^i w_2^i w_3^i ... w_n^i

Outline

1. Tools in Neural Network
2. State-of-the-Art Models for Sentiment Analysis
3. Experiment Results I
4. Adding Manifold Learning to Embedding Space
5. Experiment Results II
6. Future Work - for next 2 weeks

Experiment Settings

- All training was done on sentences broken down into elementary discourse units
  - EDUs based on Rhetorical Theory (EDU database available online)
- Loss Function: Binary Cross Entropy

$$L(\hat{y}|y^{true}) := -(y^{true}\log(\hat{y}) - (1 - y^{true})\log(1 - \hat{y}))$$

- Optimizer: Adam (adaptive momentum) optimizer
  - an exponential moving average of the gradient and the squared gradient
  - yielded about same loss, accuracy as AdaDelta, which was used in Angelidis et al. (MILNET)
  - Learning rate lr = 0.001 for all models

Accuracy Scores

Accuracy scores for various models. Model names are **bolded**.

	RNN - Benchmark	**MILNET**	**Variant**	**HierNet**
Trial 1	0.8704	0.8910	0.8900	0.8937
Trial 2	-	0.8947	-	-

Outline

1. Tools in Neural Network
2. State-of-the-Art Models for Sentiment Analysis
3. Experiment Results I
4. Adding Manifold Learning to Embedding Space
5. Experiment Results II
6. Future Work - for next 2 weeks

Post-Processing Word Embedding

As per Hasan et al 2017, re-embedding less frequent words will improve the learning process for neural networks. While the paper only attempted Locally Linear Embedding (LLE), both LLE and IsoMap were selected in an attempt to improve the accuracy.

1. Locally Linear Embedding (LLE)
2. Isometric Feature Mapping (IsoMap)

How to Re-embed Words

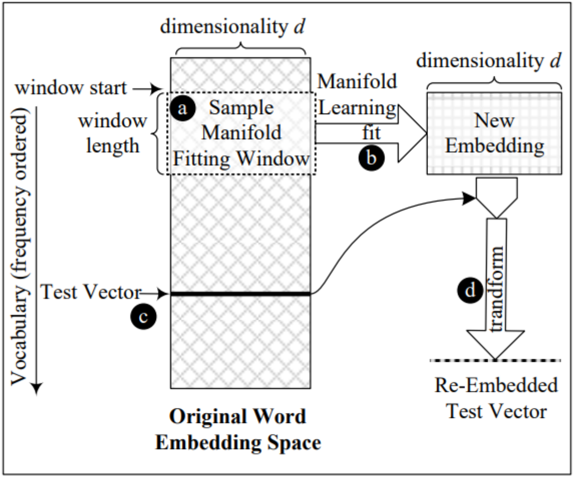


Figure 2: Re-Embedding via Manifold Learning.

Locally Linear Embedding

- **Intuition:** Assuming that for every point in the embedding space, its relationship between itself and its neighbours is linear. Hence, any re-transformation of the space should also preserve linearity (through translation, rotation, and re-scaling).

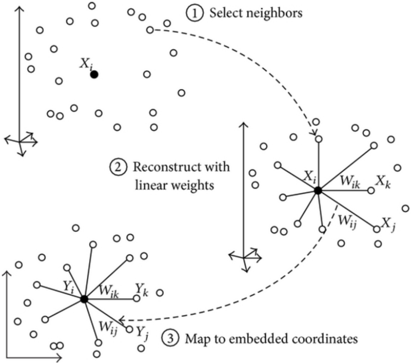


Figure: Intuition for LLE

Locally Linear Embedding

Algorithm

1. compute the k nearest neighbors for each data point
2. find linear distances matrix W st. for each element $w_{ij} in W$:

$$\begin{aligned} \min_{W} \quad & \Sigma_{i=1}^{n} \left| x_i - \Sigma_{j=1}^{k} w_{ij} * x_{v(i)_j} \right|^2 \\ \text{s.t} \quad & \Sigma_{j=1}^{k} w_{ij} = 1 \\ & w_{ij} = 0[note] \end{aligned}$$

   $[note]$ for all x_j not in k-nearest neighbourhood set and $x_{v(i)_j} = j^{th}$ neighbour of x_i

3. find transformed vector Y st. for each element y_i in Y:

$$\begin{aligned} \min_{Y} \quad & \Sigma_{i=1}^{n} \left| y_i - \Sigma_{j=1}^{k} w_{ij} * y_{v(i)_j} \right|^2 \\ \text{s.t} \quad & Y^T Y / n = 1 \\ & \Sigma_{i=1}^{n} y_i = 0 \end{aligned}$$

Isometric Feature Mapping

- **Intuition:** Euclidean distance may be a poor measure of disimilarity between points. Geodesic distance, distance between points along the manifold, can more accurately capture the neighbourhood relationships that should be preserved.

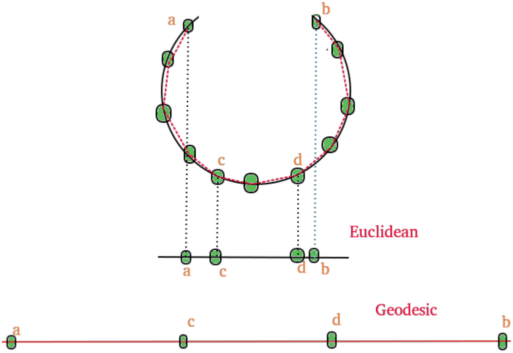


Figure: Geodesic Distance

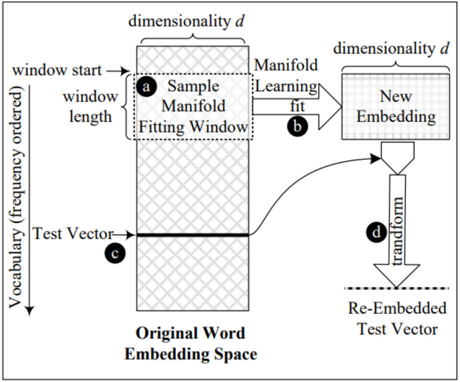
Isometric Feature Mapping

Algorithm

1. compute the k nearest neighbors for each data point
2. compute geodesic distance between all pairs of points using algorithms such as Floyd-Warshall algorithm or Floyd pipelined 2-D block algorithm.
3. use Multiple-Dimension Scaling technique to reduce dimensions.

Post-Processing Settings

- For both LLE and IsoMap, 300 (the same dimension as the initial embedding) were used for the dimension output space as per *Hasan et al 2017*.
- Only the first 1000 words were considered.
- Potential to try different dimensions and other hyperparameters.



Outline

1 Tools in Neural Network

2 State-of-the-Art Models for Sentiment Analysis

3 Experiment Results I

4 Adding Manifold Learning to Embedding Space

5 Experiment Results II

6 Future Work - for next 2 weeks

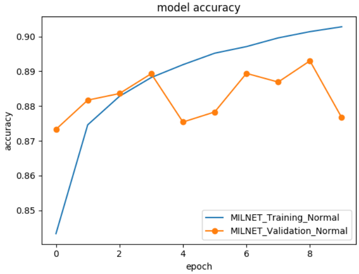
Accuracy Scores

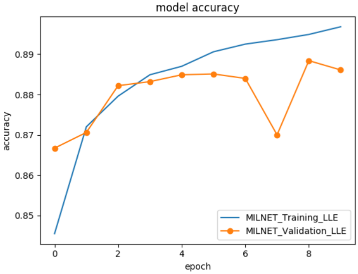
	MILNET (Trial 1)	MILNET (Trial 2)	MILNET (Trial 3 *)
Normal	0.8910	0.8947	0.8758
LLE (d = 300)	0.8842	0.8848	0.8828
IsoMap (d = 300)	0.8832	0.8759	0.8825

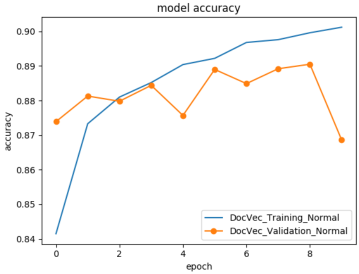
	RNN(Benchmark)	Variant	HierNet
Normal	0.8704	0.8900	0.8937
LLE (d = 300)	-	0.8842	0.8835

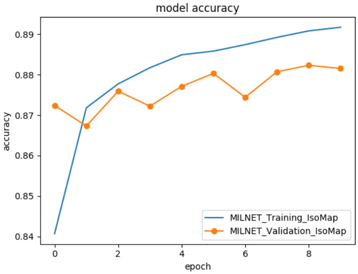
* Note: Trial 3 used a smaller polarity score dimension for the sentence output

Convergence Plots

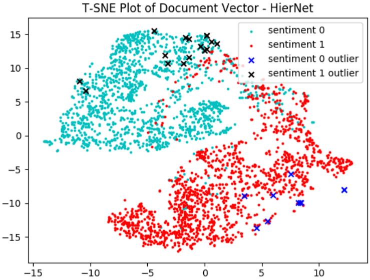




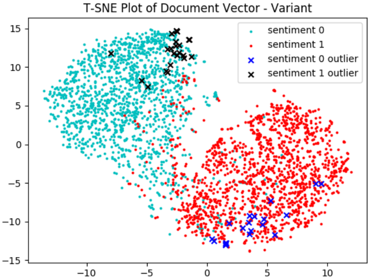




Visualization - T-SNE of Document Vectors

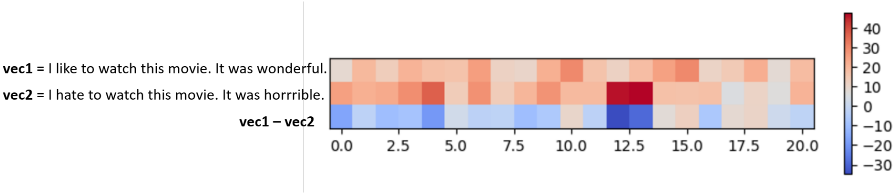


(a) Document Vectors from HierNET. Outliers are marked with 'x'

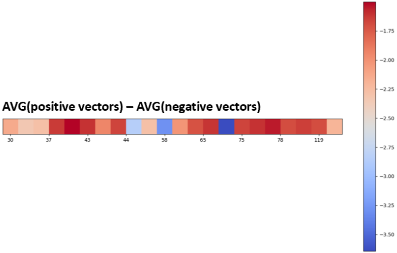


(b) Document Vectors from the Variant. Outliers are marked with 'x'

Visualization - Document Vectors of Unseen Inputs



Difference between average of positive vectors and negative vectors in testing set. Only those with absolute difference greater than 1.5 are selected.(Right)



Outline

1. Tools in Neural Network
2. State-of-the-Art Models for Sentiment Analysis
3. Experiment Results I
4. Adding Manifold Learning to Embedding Space
5. Experiment Results II
6. **Future Work - for next 2 weeks**

Future Work - for next 2 weeks

- Determine optimal size for EDU sentences
  - Gridsearch
  - Inquire with the original author
- Investigate into other techniques to visualize results
- Examine information held in transformed embedding space
- Use a unweighted summation function instead of a FFNN to produce sentiment scores from document vectors.
- Investigate ways to improve MILNET result to beyond 91%

For Further Reading I

- Stefanos Angelidis and Mirella Lapata.
  Multiple Instance Learning Networks for Fine-Grained Sentiment Analysis.
  Transactions of the Association for Computational Linguistics, vol. 6, pp. 1731, 2018.

- Souleiman Hasan and Edward Curry.
  Word Re-Embedding via Manifold Dimensionality Retention.
  Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 321326 Copenhagen, Denmark, September 711, 2017.

- Zichao Yang, Diyi Yang, Chris Dyer et al.
  Hierarchical Attention Networks for Document Classification.
  Proceedings of NAACL-HLT 2016.