Introduction to Data Science Final Project: Semantic Analysis of Yelp Dataset

Jia-Wei Liao[1], Yi-Cheng Hung[1], Yu-Lin Tsai[2]
Advisor: Horng-Shing Lu[3]

Department of [1] Applied Mathematics, [2] Arete Honors Program, [3] Statistics
National Yang Ming Chiao Tung University

September 23, 2022

Outline

Motivation

- Do not know what to eat after going out
- It is very troublesome to prepare in advance

Introduction

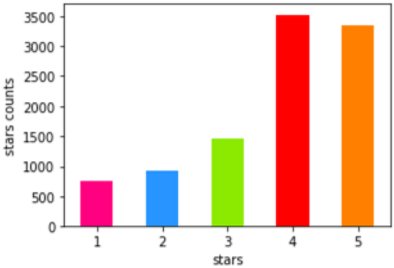
Yelp Dataset: This data set mainly collects information on restaurant reviews and satisfaction ratings.

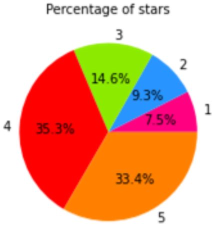
Goal: Use the customer review to analyze whether the customer is satisfied with the foods.

Data Preview

We have 10,000 samples of data at the first.

business_id	date	review_id	stars	text	type	user_id
9yKzy9PApeiPPOUJEtnvkg	2011-01-26	fWKvX83p0-ka4JS3dc6E5A	5	My wife took me here on my birthday for breakf...	review	rLtl8ZkDX5vH5nAx9C3q5Q
ZRJwVLyzEJq1VAihDhYiow	2011-07-27	IjZ33sJrzXqU-0X6U8NwyA	5	I have no idea why some people give bad review...	review	0a2KyEL0d3Yb1V6aivbIuQ
6oRAC4uyJCsJl1X0WZpVSA	2012-06-14	IESLBzqUCLdSzSqm0eCSxQ	4	love the gyro plate. Rice is so good and I als...	review	0hT2KtfLiobPvh6cDC8JQg
_1QQZuf4zZOyFCvXc0o6Vg	2010-05-27	G-WvGaISbqqaMHINnByodA	5	Rosie, Dakota, and I LOVE Chaparral Dog Park!!...	review	uZetl9T0NcROGOyFfughhg
6ozycU1RpktNG2-1BroVtw	2012-01-05	1uJFq2r5QfJG_6ExMRCaGw	5	General Manager Scott Petello is a good egg!!!...	review	vYmM4KTsC8ZfQBg-j5MWkw



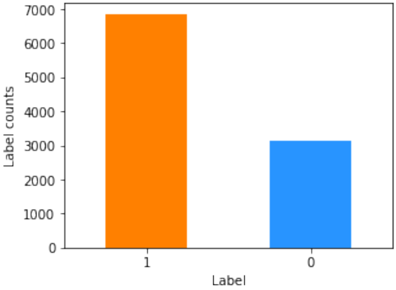


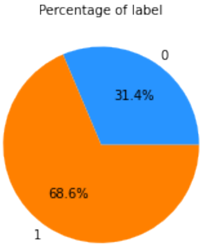
Data Preview

- Define the label as

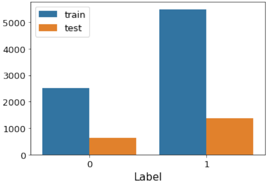
$$\text{label}_i = \begin{cases} 1, & \text{star}_i \geq 4 \\ 0, & \text{otherwise} \end{cases}$$

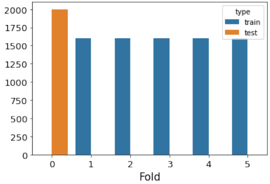
- There are 6863 data with label 1 and 3137 data with label 0

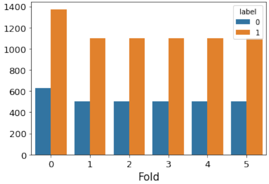


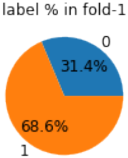


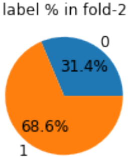
Split training set and testing set

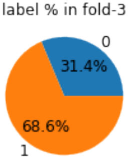


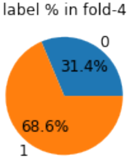


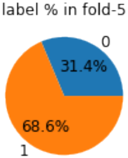












Eliminate Stop Words

Stop words are the words which are mostly used as fillers and hardly have any useful meaning. So, we use the following method:

1. Remove punctuation and uniform lowercase by ourselves
2. Natural Language Toolkit (NLTK) package

Eliminate Stop Words

Review:

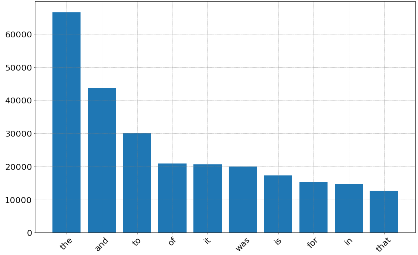
My wife took me here on my birthday for breakfast and it was excellent. The weather was perfect which made sitting outside overlooking their grounds an absolute pleasure.

Sentence:

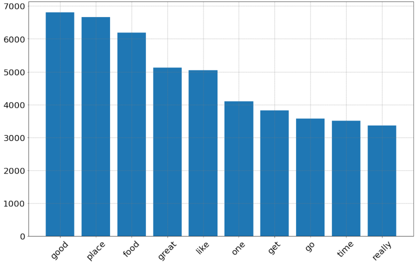
- **Self:** my wife took me here on my birthday for breakfast and it was excellent the weather was perfect which made sitting outside overlooking their grounds an absolute pleasure
- **NLTK:** wife took birthday breakfast excellent weather perfect made sitting outside overlooking grounds absolute pleasure

Eliminate Stop Words

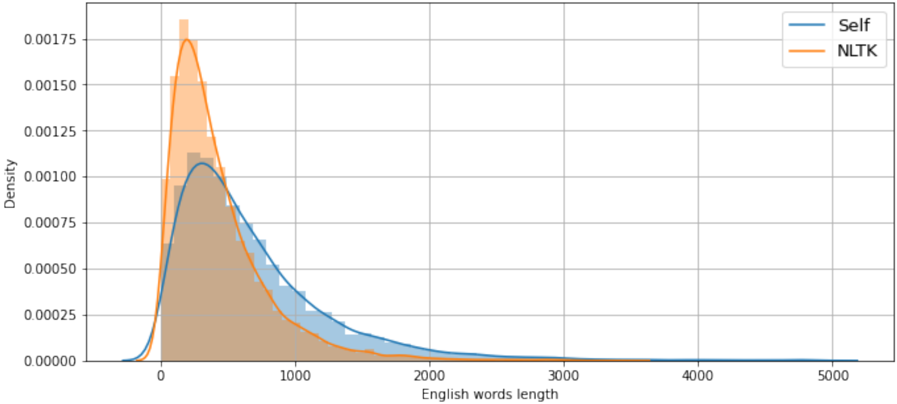
Self



NLTK



Data Preview



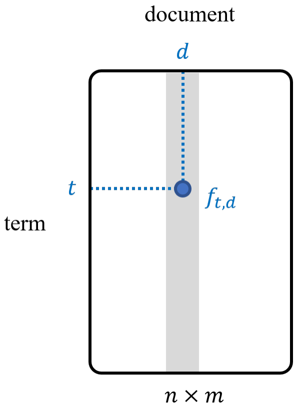
Term Frequency (TF)

Let $f_{t,d}$ be the frequency of term t in the document d.

Term frequency (TF)

Term frequency is the number of times each word appeared in document with normalization.

$$\mathrm{TF}(t,d) = \frac{f_{t,d}}{\sum_{t=1}^{n} f_{t,d}}$$



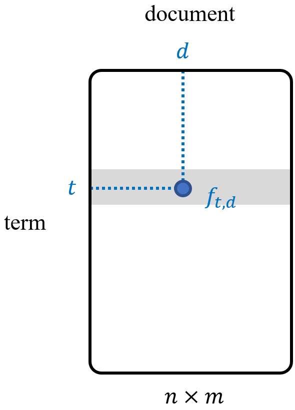
Inverse Document Frequency (IDF)

Let $f_{t,d}$ be the frequency of term t in the document d.

Inverse Document Frequency (IDF)

Document frequency is the number of documents which contain the term t. Define inverse document frequency as follow:

$$\mathrm{IDF}(t) = \log \frac{m}{1 + |\{d \mid f_{t,d} > 0\}|}$$

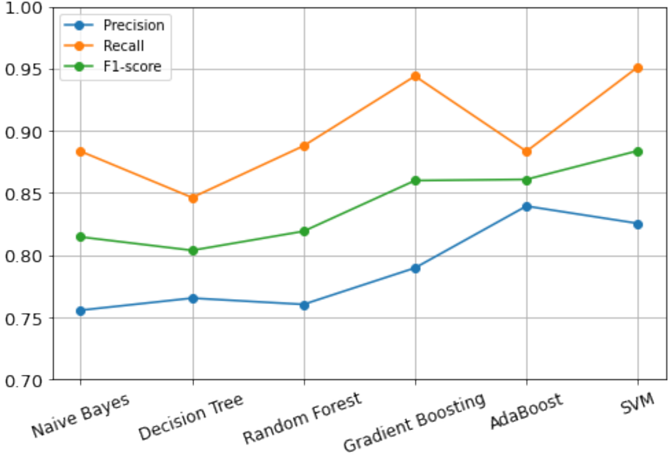


TF-IDF

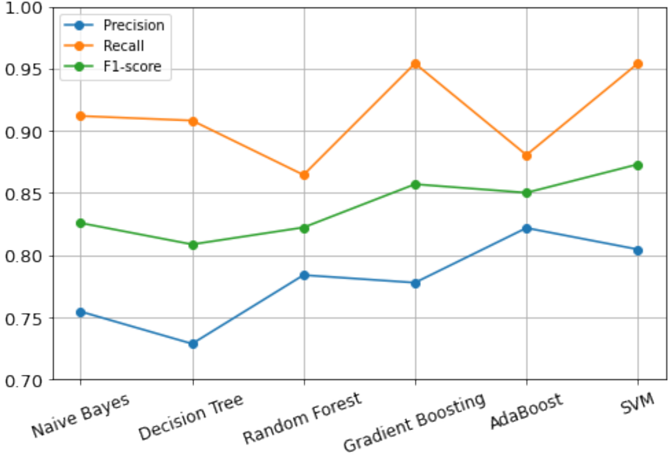
$$\text{TF-IDF}(t,d) = \text{TF}(t,d) \times \text{IDF}(t)$$

Top 5	Doc1	Doc2	Doc3	Doc4
good	0	0.0303	0.1388	0
place	0.0417	0.0298	0	0
food	0.0438	0	0	0
great	0	0.0329	0	0
like	0.0480	0.1029	0	0

Experiment: preprocessing by ourselves



Experiment: preprocessing by NLTK



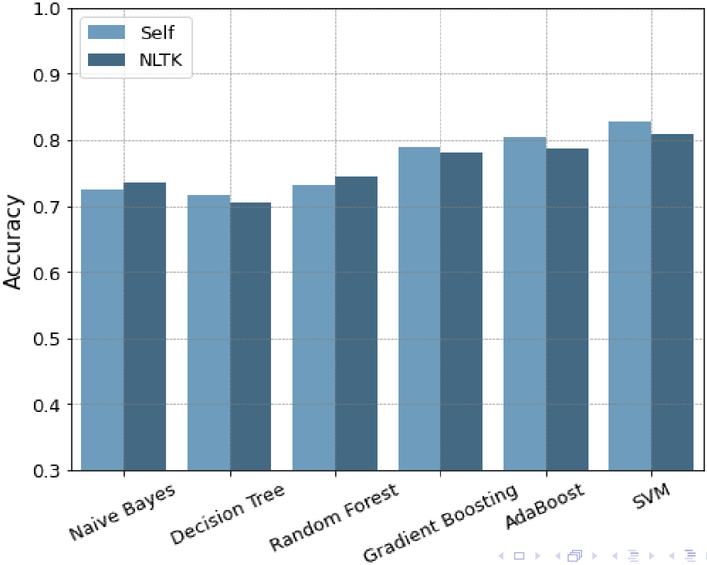
Experiment: preprocessing by ourselves

Model	Precision	Recall	F1-score	Accuracy
Naive Bayes	0.7557	0.8834	0.8016	0.7240
Decision Tree	0.7654	0.8463	0.8038	0.7165
Random Forest	0.7604	0.8878	0.8192	0.7310
AdaBoost	**0.8394**	0.8834	0.8608	0.8040
Gradient Boosting	0.7897	0.9439	0.8598	0.7890
SVM	0.8255	**0.9512**	**0.8838**	**0.8285**

Experiment: preprocessing by NLTK

Model	Precision	Recall	F1-score	Accuracy
Naive Bayes	0.7547	0.9119	0.8259	0.7360
Decision Tree	0.7288	0.9082	0.8087	0.7050
Random Forest	0.7840	0.8645	0.8223	0.7435
AdaBoost	**0.8219**	0.8806	0.8502	0.7870
Gradient Boosting	0.7779	**0.9541**	0.8570	0.7815
SVM	0.8047	**0.9541**	**0.8730**	**0.8095**

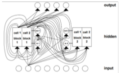
Experiment



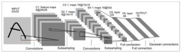
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
Accuracy
Self
NLTK
Naive Bayes
Decision Tree
Random Forest
Gradient Boosting
AdaBoost
SVM

History of Deep Learning in NLP

1997 — LSTM



1998 — CNN



2013 — Word to vector

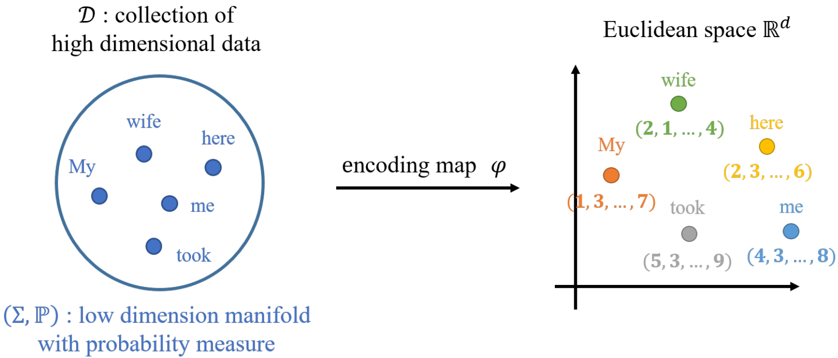
2017 — Transformer

2018 — BERT

Date Manifold

Manifold Assumption

Natural high dimensional data concentrates close to a non-linear low-dimensional manifold.



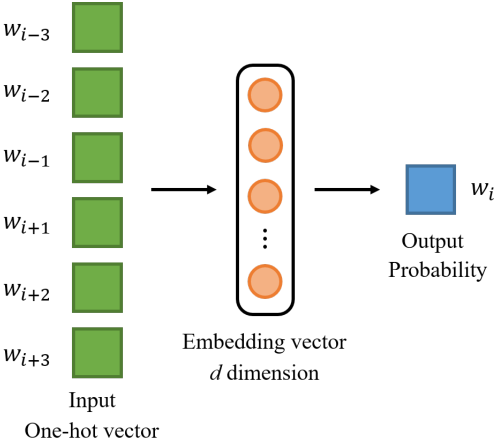
Word Embedding Model

- **Count-Based:** TF-IDF
- **Prediction-Based** [1] **:** CBOW, Skip gram

[1] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, Computation and Language, 2013

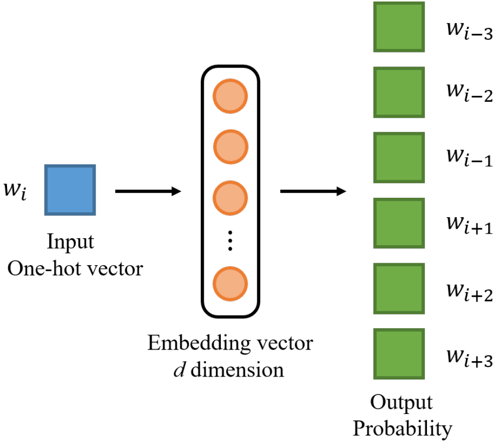
Word Embedding Model

Continuous bag of word (CBOW)



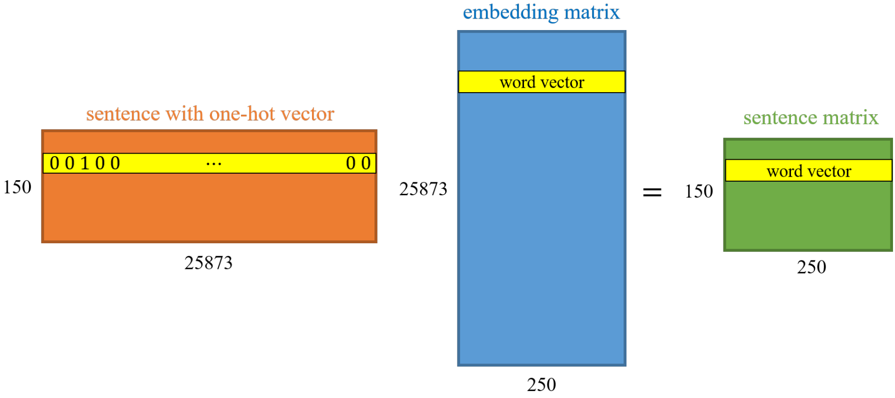
Word Embedding Model

Skip Gram



Embedding Matrix

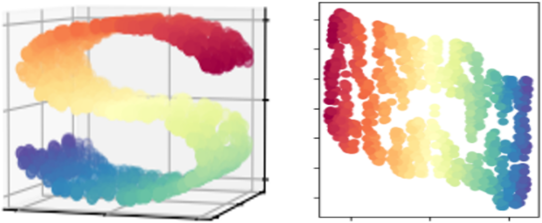
- number of the word: 25873
- maximum length: 150
- embedding dimension: 250



Visualizing Data Using t-SNE

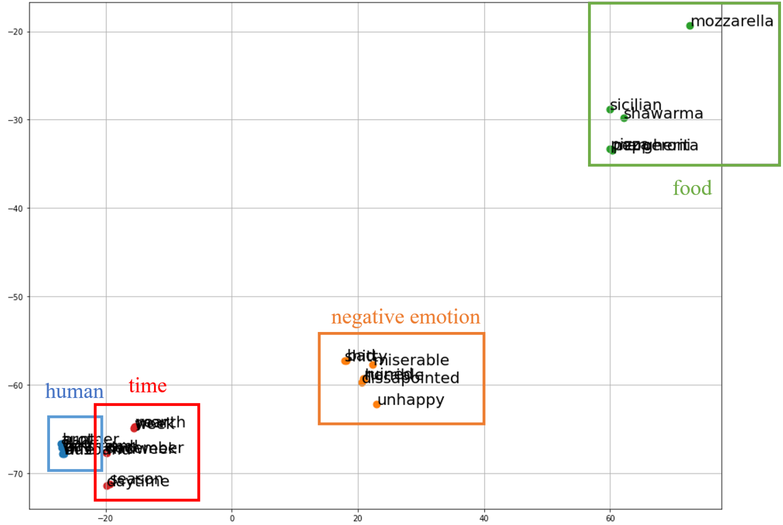
T-distributed Stochastic Neighbor Embedding (t-SNE) [2]

- It's a manifold learning
- It converts similarities between data points to joint probabilities and minimize the KL divergence

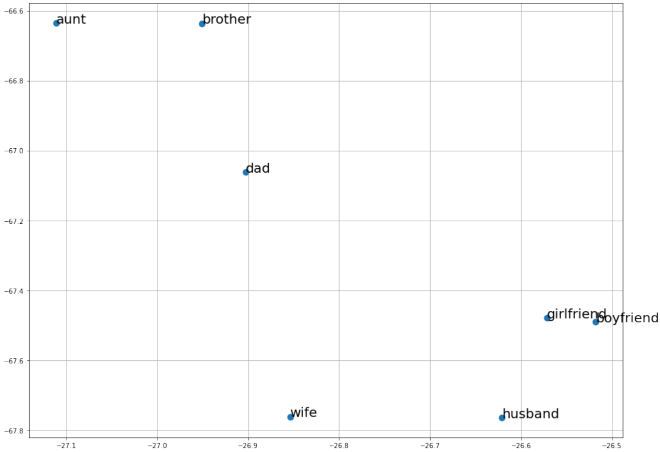


[2] L.-M., G. Hinton, Visualizing Data usingt-SNE, Journal of Machine Learning Research, 2008.

Visualizing Data Using t-SNE

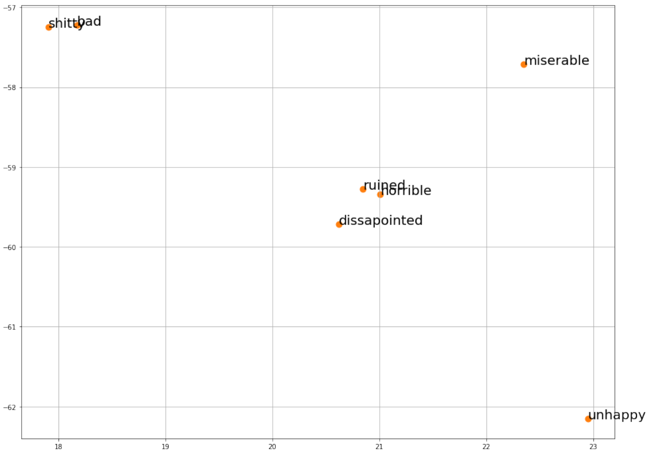


Visualizing Data Using t-SNE



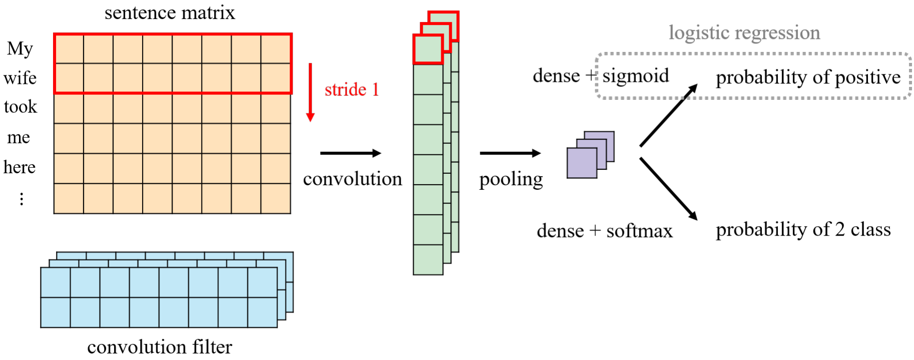
wife similar word	husbands	girlfriend	boyfriend	dad	brother
cosine similarity	0.77	0.75	0.74	0.71	0.70

Visualizing Data Using t-SNE

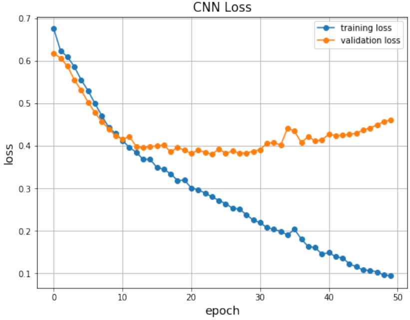


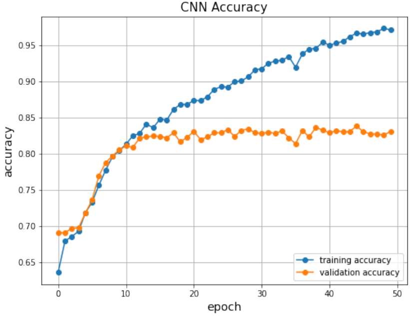
bad similar word	horrible	shitty	ruined	dissapointed	unhappy
cosine similarity	0.64	0.63	0.60	0.60	0.60

Convolutional Neural Network (CNN)



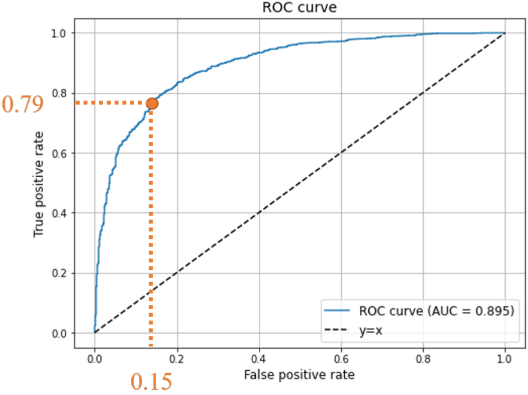
Experiment: CNN





Model	Precision	Recall	F1-score	Accuracy
CNN with sigmoid	0.8317	0.9006	0.8648	0.8215
CNN with softmax	0.9038	0.8558	0.8792	0.8295

Evaluation: ROC Curve



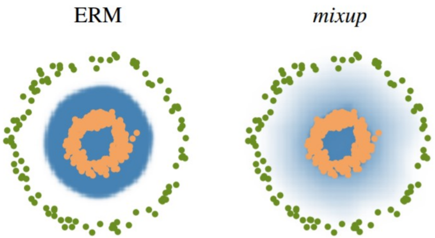
Model	Thresholds	Accuracy
CNN + sigmoid	0.5	0.8215
CNN + sigmoid	0.33 (best)	0.8315 (+0.01)

Data Augmentation

Mixup [3]

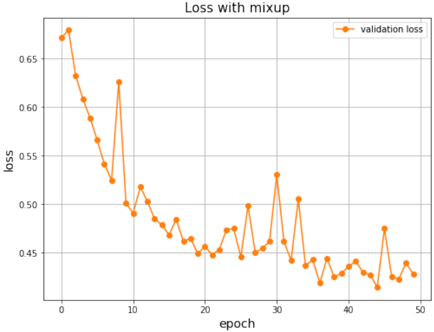
Given $(x_i, y_i), (x_j, y_j) \in \mathcal{D}$ and $\lambda \sim \text{Beta}(\alpha, \alpha)$ with $\alpha \in (0, \infty)$

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j,$$
$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j$$



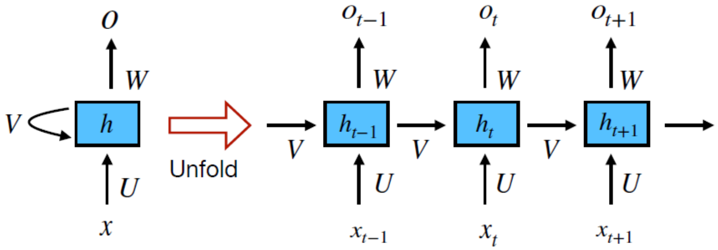
[3] H. Zhang, M. Cisse, Y.-N. Dauphin, and D. Lopez-Paz mixup: Beyond Empirical Risk Minimization, ICLR, 2018.

Experiment: CNN + Mixup



Model	Accuracy
CNN + sigmoid (thresholds = 0.5)	0.8215
CNN + sigmoid (thresholds = 0.33)	0.8315 (+0.010)
CNN + softmax	0.8295
CNN + softmax + **mixup**	0.8320 (+0.002)

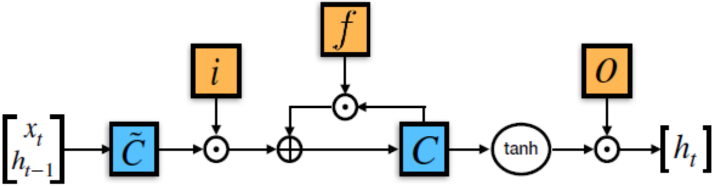
Recurrent Neural Network (RNN)



$$s_t = f(Ux_t + Ws_{t-1})$$

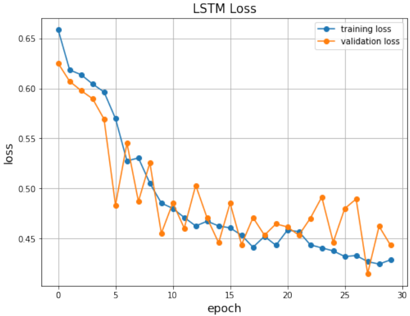
- s_t is calculated based on the current input and the previous time step's hidden state.
- f is non-linear transformation

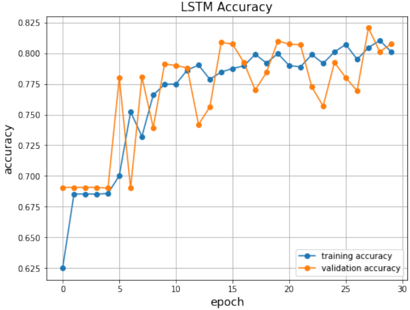
Long Short Term Memory (LSTM)



$$
\begin{aligned}
x &= [h_{t-1} \quad x_t]^\top \\
f_t &= \sigma(W_f x + b_f) \\
i_t &= \sigma(W_i x + b_i) \\
o_t &= \sigma(W_o x + b_o) \\
c_t &= f_t \odot c_{t-1} + i_t \tanh(W_c X + b_c) \\
h_t &= o_t \odot \tanh c_t
\end{aligned}
$$

Experiment: LSTM





Finally, We get the test accuracy of 0.8130.

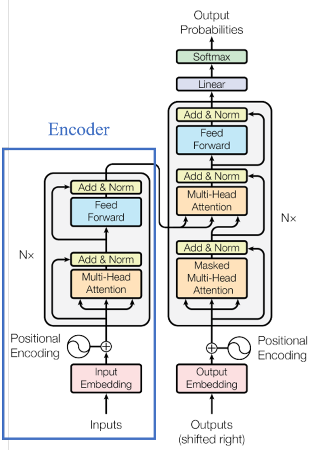
Transformer

Transformer

BERT

Bidirectional Encoder Representations from Transformers

BERT : Encoder of Transformer [4]

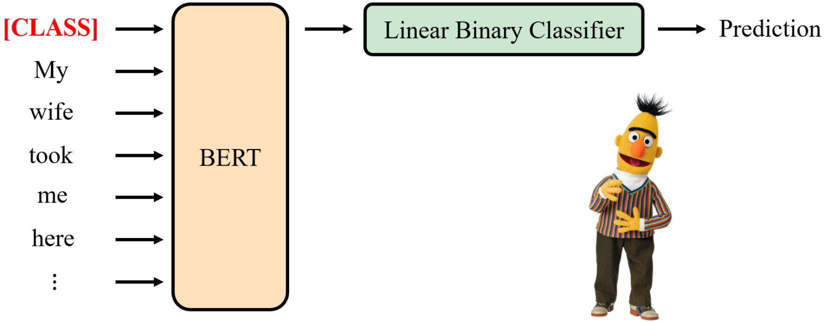


[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.-N. Gomez, L. Kaiser, I. Polosukhin, Attention Is All You Need, Computation and Language, 2017.

BERT

We use pretrain weight in the BERT and connect the linear binary classifier at the end.

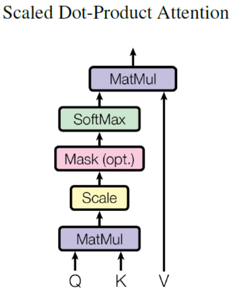
- Input: sentences
- Output: predicted class

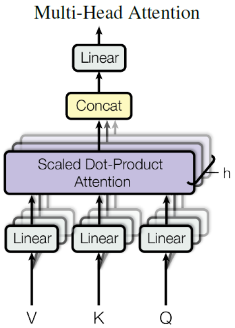


Attention Block

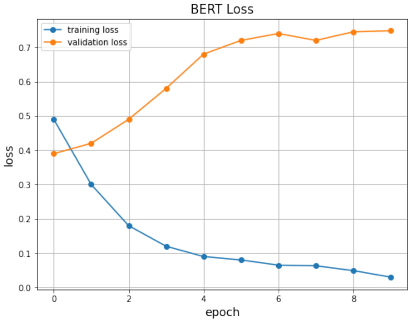
- Q: queries, K: keys, V: values
- d_k is keys of dimension

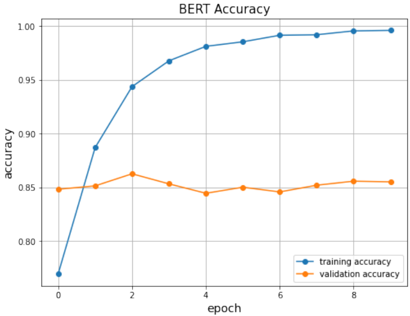
$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$





Experiment: BERT





Finally, We get the test accuracy of 0.8550.

Conclusion

In this project, we implement machine learning methods and deep learning methods. The deep learning model gets good performance. We compared the result as follow:

- **Machine Learning method:**

Method	Naive Bayes	Tree-based	SVM
Accuracy	0.7240	0.8040	0.8285

- **Deep Learning method:**

Model	CNN (with mixup)	LSTM	BERT
Accuracy	0.8320	0.8130	0.8550

Conclusion

In this project, we implement machine learning methods and deep learning methods. The deep learning model gets good performance. We compared the result as follow:

- **Machine Learning method:**

Method	Naive Bayes	Tree-based	SVM
Accuracy	0.7240	0.8040	0.8285

- **Deep Learning method:**

Model	CNN (with mixup)	LSTM	BERT
Accuracy	0.8320	0.8130	0.8550

Machine learning methods are more explainable but deep learning method like black boxes. At the recent, many research start to develop **Explainable AI**. So, we can develop towards this research topic in the future.

Reference

1. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Computation and Language, 2018.
2. Laurens van der Maaten, Geoffrey Hinton, Visualizing Data using t-SNE, Journal of Machine Learning Research, 2008.
3. Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, Efficient Estimation of Word Representations in Vector Space, Computation and Language, 2013.
4. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, Attention Is All You Need, Computation and Language, 2017.
5. Tom Young, Devamanyu Hazarika, Soujanya Poria, Erik Cambria, Recent Trends in Deep Learning Based Natural Language Processing, Computation and Language, 2017.
6. Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, David Lopez-Paz, mixup: Beyond Empirical Risk Minimization, ICLR, 2018.

Thanks for listening!