Just *One Shot* of Wine?
Developing Siamese LSTM Models for Sentence Similarity

Tatiana Wu, Tom Knowles
{twu99}/{tknowles}@stanford.edu

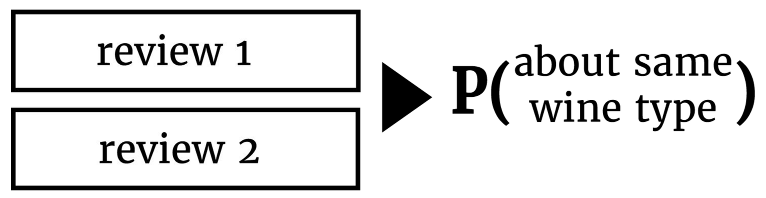
CS229: Machine Learning
Stanford University, Spring 2019

Introduction

- We consider the general problem of sentence similarity, which we treat as a textual classification problem.
- This has application in a number of fields: comment deanonymization, intent recognition, chat-bots, and web parsing.

Problem

- We assume that given two wine reviews, we return a prediction/probability. of whether they're about the same wine
- This is easier if we have all the labels for the possible wines, since we can predict for each label.



- Accuracy also varies based off whether the wine categories are included in the training (interpolating), or are not (extrapolating).

Data

Input data:

- Wine Review Dataset from Kaggle [1]
- >100,000 reviews of wines

Pre-processsing:

- Restrict to the top 50 most common wine categories, each of which has over 200 reviews.
- We remove words that may make the task too easy—like label names.
- Somewhat balance the dataset.

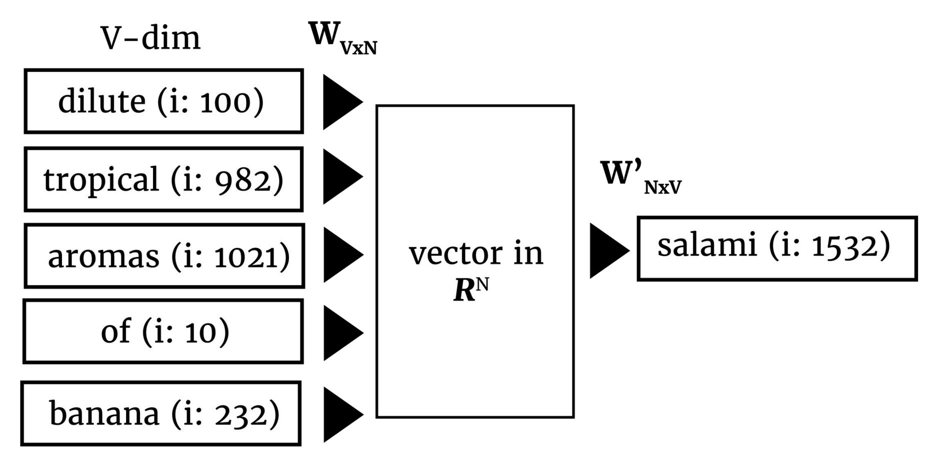
Final dataset:

- 100,000 comments, which are fairly evenly distributed among the top.
- We sample random pairs, so interpolate.

Word Embeddings

We use word embeddings as our core feature extraction method.

- We create 300-dimensional vectors representing each word.
- 300 dimensions was chosen as this is industry-standard.
- We try pre-trained GLoVE and word2vec vectors, and word2vec vectors trained our data with a CBOW model.



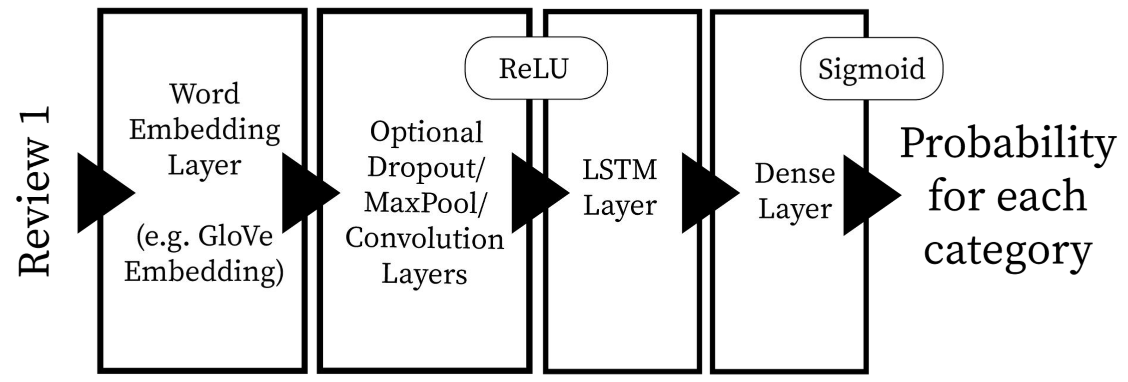
Setting 1

Problem definition:

- We can enumerate all the wine categories.
- We train a model to predict a single wine category, and then predict probability using its outputs.

Methodology:

- We use this neural network architecture, with cross-entropy loss:



- We also implement simple baselines, simply averaging word vectors (a "bag of words" model) to turn sentences into features. These include:
- Support Vector Machines with RBF Kernel.
- Multiple Logistic Regression Models.

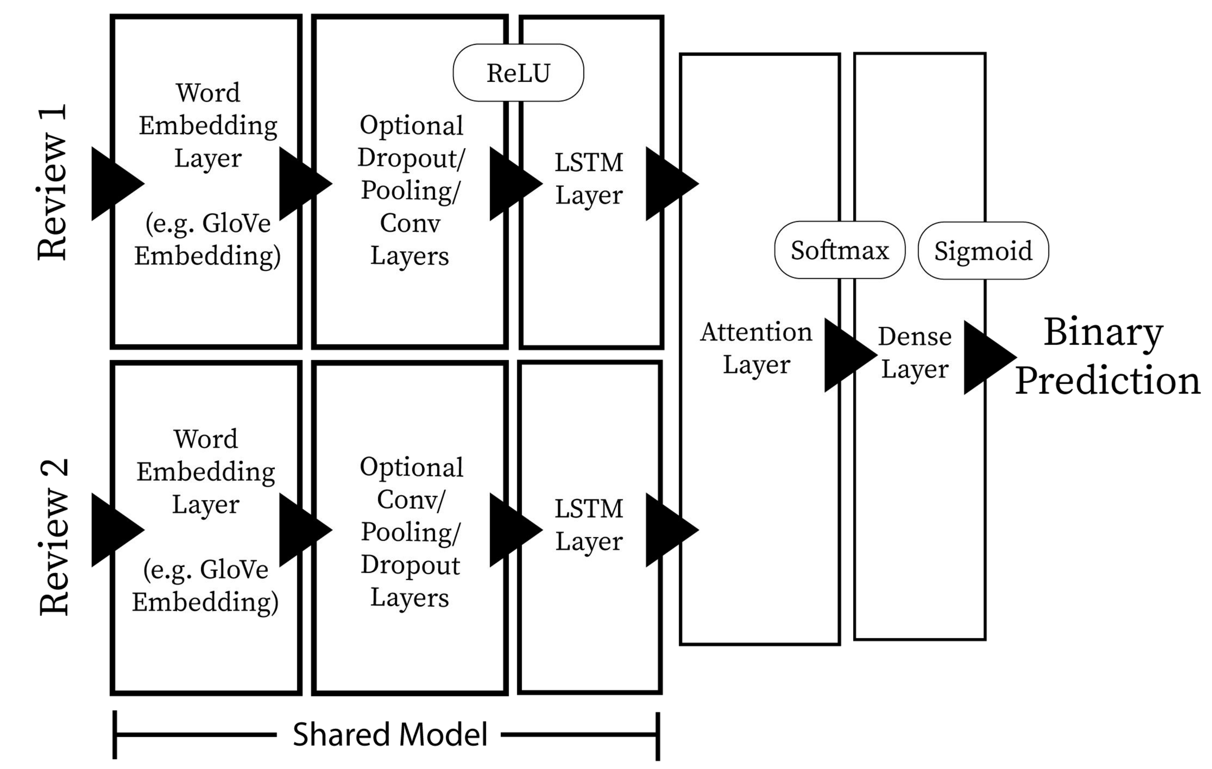
Setting 2

Problem definition:

- We cannot enumerate all categories.
- We simply train a model to compare pairs of reviews—a type of "one-shot" learning.

Architecture:

- We use the following Siamese architecture, with a shared model for the branches.



- The attention layer assigns weight to all the LSTM hidden states against a target vector u_w.

1. $u_{it} = \tanh(W_w h_{it} + b_w)$
2. $\alpha_{it} = \frac{\exp(u_{it}^\top u_w)}{\sum_t \exp(u_{it}^\top u_w)}$
3. $s_i = \sum_t \alpha_{it} h_{it}$

- We also try a variation of this model, replacing last layers with a Manhattan Distance model (only final LSTM state.)

Discussion

Significant success when we have all labels

- Addition of convolutional layers allows it to generalise better
- Accuracy significantly above baseline

One-shot learning is significantly harder

- Unbalanced dataset (many more dissimilar pairs than similar pairs)
- Model is very slow to train, due to large dataset and parameter count

Results

We look at F1 score of the binary classifier—(directly comparable to multi-class accuracy for Setting 1)—random baseline is 10%.

Setting 1:

		Logistic Regression (Bag of Words)	SVM (Bag of Words)	LSTM Conv. NN
GLoVe (pre-trained)	*LRAP of single classification*	55.2%	49.8%	75.0%
	F1 Score of pair classification	39.0%	29.4%	66.4%
Word2Vec (trained on data)	*LRAP of single classification*	65.8%	64.0%	79.6%
	F1 Score of pair classification	50.0%	47.2%	70.0%

Setting 2:

		Manhattan Distance	Neural Net Without Convolution	Neural Net With Convolution
GLoVe	*Train*	–	–	65.0%
	Test	–	–	26.6%
Word2Vec	*Train*	40.1 %	75.1%	68.8%
	Test	23.2%	26.7%	30.8%

Future Work

- Much better GPUs are needed to train such a large model.
- Siamese Model fails to generalise in a fairly small number of epochs—it may have too many parameters.
- Change dataset to be more relevant to sentence similarity; we realised that a lot of reviews are not similar sentences.

References

1. https://www.kaggle.com/zynicide/wine-reviews
2. Pennington, Jeffrey, et al. "Glove: Global Vectors for Word Representation." *Empirical Methods in Natural Language Processing (EMNLP)*, 2014,.
3. Mueller, Jonas and Thyagarajan, Aditya. "Learning Sentence Similarity with Siamese Recurrent Architectures." *AAAI.* 2016
4. Chi, Ziming and Zhang, Bingyan. "A Sentence Similarity Estimation Method Based on Improved Siamese Network", *Journal of Intelligent Learning Systems.* 2018.