Transformers and User Intent

Brandon Szeto
Jacobs School of Engineering
University of California, San Diego
San Diego, CA 92122
bszeto@ucsd.edu

Darren Yu
Jacobs School of Engineering
University of California, San Diego
San Diego, CA 92122
dmyu@ucsd.edu

Nathaniel Thomas
Jacobs School of Engineering
University of California, San Diego
San Diego, CA 92122
nathomas@ucsd.edu

Abstract

Today, large language models have become immensely popular. We use one such model, BERT, for user intent classification, leveraging the Amazon Massive Intent Dataset. BERT, a state-of-the-art transformer architecture, has demonstrated remarkable performance across various natural language processing tasks. We fine-tune BERT for user intent classification and then explore advanced training techniques to enhance its performance. Additionally, we investigate the application of contrastive losses for further improvements in model accuracy. We preprocess the dataset, fine tune BERT, implement custom training techniques derived from recent research, and experimenting with contrastive learning methods. Through this process, we aim to provide insights into the efficacy of different strategies for enhancing user intent classification tasks. WE NEED PERCENT CORRECT NUMBERS HERE.

Preprint. Under review.

Introduction

Related Works

BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding

In Devlin et al. [2019] the authors introduce a groundbreaking model in natural language processing that utilize the concept of bidirectional context representation learning. By pretraining a transformer-based neural network on large corpora with masked language modeling and next sentence prediction tasks, BERT is able to capture deep contextual information from both left and right contexts of a word. This bidirectional understanding of text enables BERT to achieve state-of-the-art performance on a wide range of NLP tasks, including but not limited to question answering, sentiment analysis, named entity recognition, and machine translation.

Supervised Contrastive Learning

In Khosla et al. [2021] the authors introduce supervised contrastive learning, a method for learning representations of data by maximizing agreement between similar pairs and minimizing agreement between dissimilar pairs. Supervised contrastive learning leverages labeled data to guide the learning process, making it particularly suitable for tasks where labeled examples are abundant. The method has shown promising results in various applications, including image classification, natural language processing, and speech recognition.

SimCSE: Simple Contrastive Learning of Sentence Embeddings

In Gao et al. [2022], the authors propose a simple yet effective approach for learning sentence embeddings through contrastive learning. By formulating the contrastive objective in a straightforward manner, SimCSE achieves competitive performance on various downstream tasks without requiring complex architectures or extensive hyperparameter tuning. The method has been shown to learn semantically meaningful representations of sentences, making it valuable for tasks such as semantic textual similarity, text classification, and paraphrase detection.

Methods

(a) It has at least 3 subsections for baseline/custom/contrastive learning. Each of the subsection should start with the final experimental settings you decide that we do not specify, including parameters of the library functions (except for defaults), hyperparameters you choose etc. These experimental setting should correspond to your best results.

Baseline

Custom

Contrastive

Results

Summarize and document the results for all experiments (Baseline Model, Custom Finetuning Strategies, SupContrast and SimCLR).

(a) Include the table(s) (for the 6 experiments as indicated above with “exp idx”)

Exp. Index	Experiment	Loss	Accuracy
1	Test set before fine-tuning		
2	Test set after fine-tuning		
3	Test set with first technique		
4	Test set with second technique		
5	Test set with two techniques		
6	Test set with SupContrast		
7	Test set with SimCLR		

Table 1: Experimental Results

(b) Provide your training and validation accuracy in each plot for the following 5 models

i. Baseline model

ii. first fine-tuning technique

iii. second fine-tuning technique

iv. SupContrast

v. SimCLR

Discussion

Q1: If we do not fine-tune the model, what is your expected test accuracy? Explain Why.

A1: We expect a low test accuracy. Without fine tuning, we fail to leverage the general representations of language learned by the BERT model for our specific task. BERT has the potential to perform very well on our dataset, but was not trained to do so and would be limited by its generic pre-trained representations.

Q2: Do results match your expectation(1 sentence)? Why or why not?

A2:

Q3: What could you do to further improve the performance?

A3:

Q4: Which techniques did you choose and why?

A4:

Q5: What do you expect for the results of the individual technique vs. the two techniques combined?

A5:

Q6: Do results match a real niggas expectation(1 sentence)? Why or why not?

A6:

Q7: What could you do to further improve the performance?

A7:

Q8: Compare the SimCLR with SupContrast. What are the similarities and differences?

A8: Both models utilize contrastive learning to learn semantic representations. However, the models differ in complexity and implementation and have different applications. For example, SimCLR aims to create a simple and efficient framework for sentence embeddings. On the other hand, SupContrast takes a more robust approach of implementing contrastive learning for general-purpose tasks.

Q9: How does SimCSE apply dropout to achieve data augmentation for NLP tasks?

A9: By applying dropout, SimCSE introduces noise to the model, encouraging the model to learn more robust features. This effectively creates different perspectives of the same input sentence embedding, simulating the augmentation of data. The model then learns to maximize the similarity between augmented versions of the same embeddings while minimizing the similarity between augmented versions of different embeddings.

Q10: Do the results match your expectation? Why or why not?

A10:

Q11: What could you do to further improve the performance ?

A11:

Contributions

Brandon Szeto: Dataloader, tokenizer, baseline model

Darren Yu:

Nathaniel Thomas:

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. Simcse: Simple contrastive learning of sentence embeddings, 2022.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning, 2021.