Fine Tuning Language Models for Classification

Gavin Butts, Ava Hoeger

TABLE OF CONTENTS

01
Background

Data Processing

What is **Subword Tokenization**?

- Converts complex sentences into a simpler format
- Words are broken into their components (prefix, suffix, etc.)

I love machine learning! ⇒ [“i”, “love”, “machine”, “learn”, “-ing”, “!”]

Data Processing

What is **Subword Tokenization**?

- Converts complex sentences into a simpler format
- Words are broken into their components (prefix, suffix, etc.)

$$\text{I love machine learning!} \Rightarrow [\text{“i”}, \text{“love”}, \text{“machine”}, \text{“learn”}, \text{“-ing”}, \text{“!”}]$$

How do Neural Networks **process tokens**?

- Converts tokens to integers
- Input integers into model

$$[\text{“i”}, \text{“love”}, \text{“machine”}, \text{“learn”}, \text{“-ing”}, \text{“!”}] \Rightarrow [101, 806, 1143, 964, 345, 3256]$$

Language Models

What is a **Language Model**?

- A neural network that models language
- Capable of understanding relationship between words

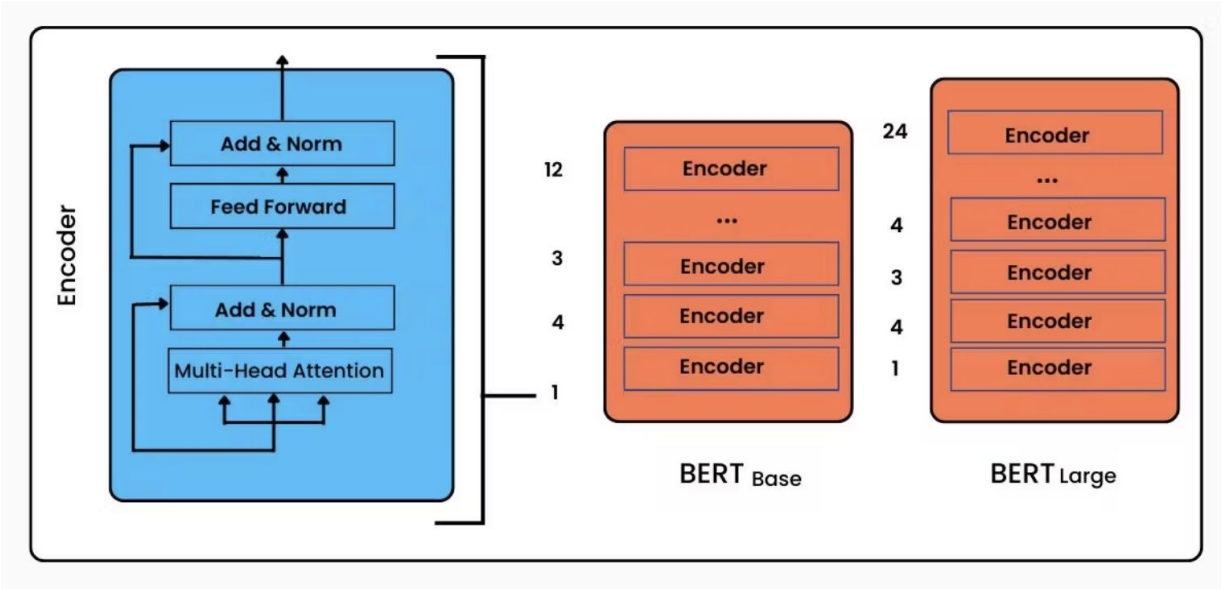
What makes the Language Models so effective?

- The building block of these models is the **Transformer**
- Transformers models how words to depend upon other words

BERT Models

BERT is a specific type of language model

- **B**idirectional **E**ncoder **R**epresentations from **T**ransformers



RoBERTa

RoBERTa is a specific type of BERT model

- **R**obustly **o**ptimized **BERT a**pproach
- Same architecture as BERT
- Trained by removing particular training steps while increasing dataset size

Full Fine Tuning

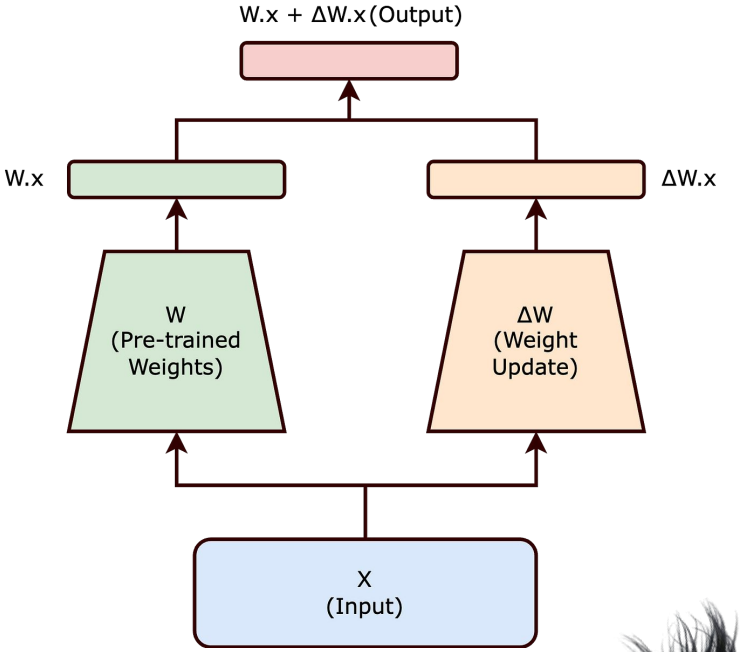
To **Fully Fine Tune** a model:

1. Identify what problem you hope to solve
2. Find an appropriate, pre-trained model
3. Find dataset associated with your problem
4. Train model with already instantiated weights

Fine Tuning via LoRA

What is **LoRA**

- **Low Rank Adaption**
- Finetunes Weight Update matrices
- Freezes Pre-trained Weights



Goal

Given a pre-trained RoBERTa model, does full fine tuning or LoRA perform better on classification tasks?

02
Dataset

Stanford Natural Language Inference (NLI)

- Stanford Natural Language Inference (SNLI), also known as Recognizing Textual Entailment (RTE), is the task of determining the inference relation between two (short, ordered) texts: *entailment*, *contradiction*, or *neutral*

- 570k human-written English sentence pairs manually labeled for balanced classification

Data Labels

For some text A and text B

- **Entailment**: Given text A, text B is clearly true
- **Neutral**: Given text A, text B could be true but we can't be certain
- **Contradiction**: Given text A, text B is definitely false

Example: *Two dogs are running through a field.*

- **Entailment**: *There are animals outdoors.*
- **Neutral**: *Some dogs are playing fetch.*
- **Contradiction**: *The dogs are sitting on a couch.*

Additional Examples

Text	Judgments	Hypothesis
A man inspects the uniform of a figure in some East Asian country.	contradiction C C C C C	The man is sleeping
An older and younger man smiling.	neutral N N E N N	Two men are smiling and laughing at the cats playing on the floor.
A black race car starts up in front of a crowd of people.	contradiction C C C C C	A man is driving down a lonely road.
A soccer game with multiple males playing.	entailment E E E E E	Some men are playing a sport.
A smiling costumed woman is holding an umbrella.	neutral N N E C N	A happy woman in a fairy costume holds an umbrella.

Data Pre-Processing

Data points with low Inter-Rater Reliability removed (where labelers disagreed):

- 0.14% of training data points removed (785/549367)
- 0.18% of testing data points removed (176/9824)

Labels re-assigned:

- entailment → 0
- neutral → 1
- contradiction → 2

Data Pre-Processing

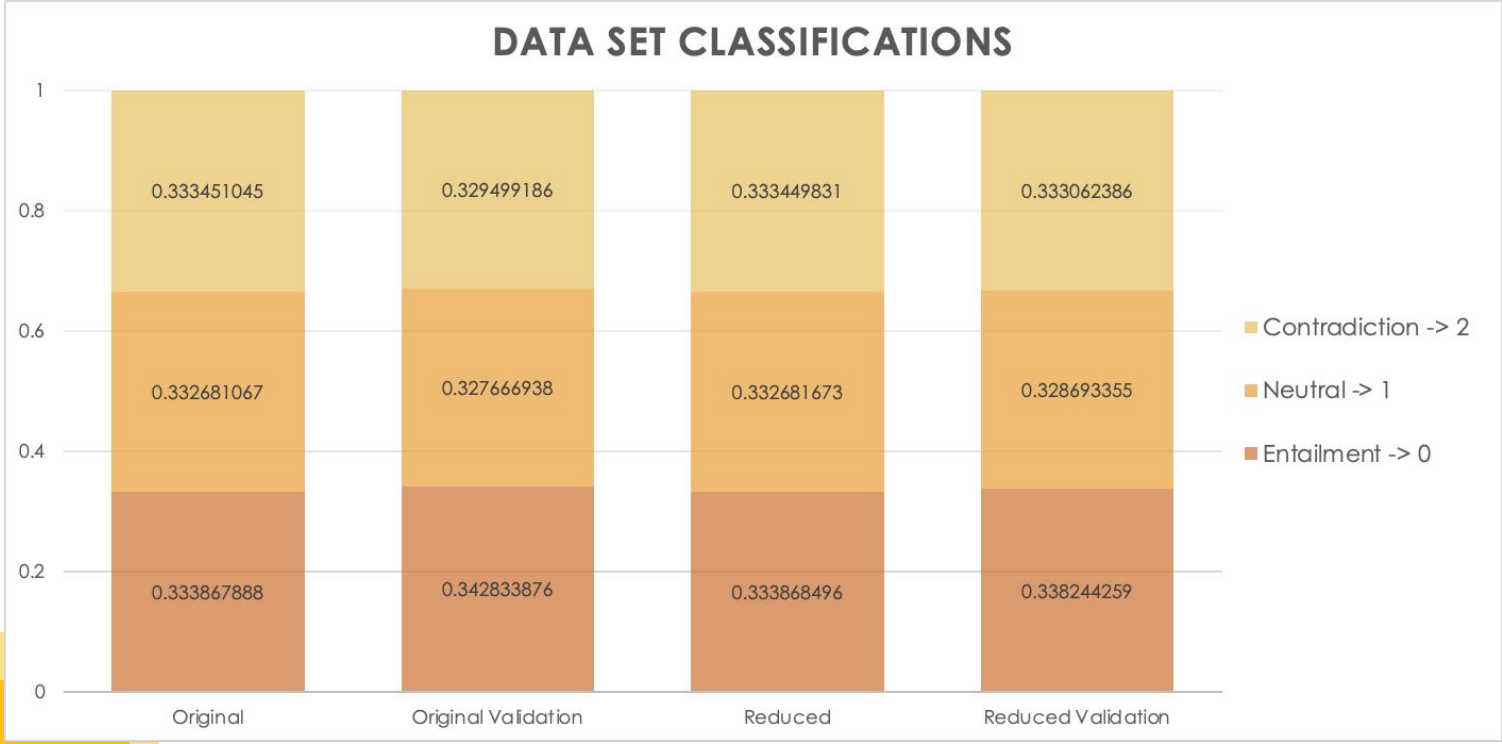
Reduced size of dataset

- Fine Tuning is usually used on tasks without access to large datasets
- Fine Tuning is usually used because of its ability to run quickly

Data set reduced by 50% via **Stratified Sampling**

- Samples data points with respect to class
- Maintains class proportions

Class Balance Original vs.



03
Results

Evaluation Metrics

Table 1: Model Performance Comparison

Method	Accuracy	Precision	Recall	F1 Score
Baseline	0.3382	0.1144	0.3382	0.1710
Full Fine-Tuning	**0.9065**	**0.9066**	**0.9065**	**0.9065**
LoRA	0.8815	0.8814	0.8815	0.8814

04
Inference

Inference Examples

On Google Colab

Examination of Low Inter-Rater Reliability

On Google Colab

Adversarial Prompting

On Google Colab

THANKS!

Questions?