# Submission and Formatting Instructions for International Conference on Machine Learning (ICML 2019)

**Anonymous Authors**[1]

## Abstract

Recent latent semantics analysis are based on calculating the word vector distance between target and goal sentences or proposing generalized semantic hypothesis according to the original text. In this project, we explore the higher level mission on challenging one of the COPA task which is analysing common sense from ordinary scenario description text. We successfully boost our BERT model performance with test accuracy of 78% among IBM research AI, BERT-mtl model with accuracy of 73.8% on the SuperGlue leaderboard 2.0. During the enhancement of model, we conjecture that the key factors in model performance is that (1) The word embedding format. (2) The scale of our fine tuning data. (3) The configuration of BERT model Introduction. We demonstrated the potential of BERT model. Experiments shows new state-of-the-art of BERT model results on COPA without any bells and whistles.

## 1. Introduction

The motivation of this project is to analyze the common sense from choice of plausible alternatives where each question is composed of a premise and two alternatives, and the task is to select the alternative that more plausibly has a causal relation with the premise. Heres the data format:

- **Premise**: The item was packaged in bubble wrap.
- **Alternative1**: It was fragile.
- **Alternative2**: It was small.
- **Question type**: Cause
- **Label**: Alternative1

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

The milestones below will show how we studied the COPA data and how we get the reasonable prediction based on the data format:

- Calculating Manhattan distance between premises and complete sentences.
- Using BERT model to calculate the euclidean distance between sentences.

### 1.1. Siamese Manhattan LSTM

**Manhattan LSTM models** has two networks $LSTM_{left}$ and $LSTM_{right}$ which process one of the sentences in a given pair independently. And a version of Manhattan LSTM where both $LSTM_{left}$ and $LSTM_{right}$ have same tied weights such that $LSTM_{left} = LSTM_{right}$. The model uses an LSTM to read in word-vectors representing each input sentence and employs its final hidden state as a vector representation for each sentence. Subsequently, the similarity between these representations is used as a predictor of semantic similarity.

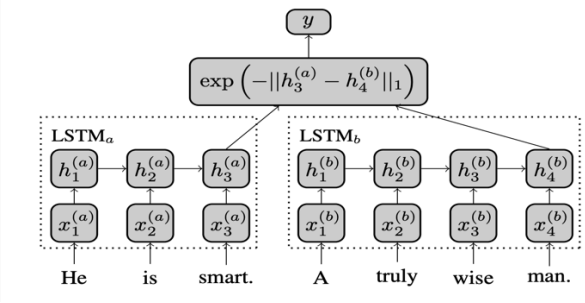


*Figure 1.* Siamese Manhattan LSTM

Here's our initial thought on how to use Siamese Manhattan LSTM model.

- Instead of initializing the Siamese networks on our test data (500 questions combined with two answers for each) to directly calculate distance between query and answer, we decide to pre-train the Siamese Manhattan LSTM with SICK data which contains 9,927

sentence pairs with a $5,000/4,927$ training/test split ( Each pair is annotated with a relatedness $label \in [1, 5]$ corresponding to the average relatedness judged by 10 different individuals).

- Then we would use our Siamese Manhattan LSTM trained from step 1 as our classifier to calculate the Manhattan distance between each query with their corresponding answer.

- We would pre-process our data based on cause & effect model, which means we would combine question and answer according to the model as our new data format, then we compare the two options in one query:
$If\ Condition = cause, data1 = query1 + answer1, data2 = query1 + answer2$
$If\ Condition = effect, data1 = answer1 + query1, data2 = answer2 + query2$

- The Loss can be conclude into three part $loss_{pretrainedmodel}, loss_{SimilarityCalculation}$ and $loss_{Classification}$

**Discussion:** According to our research on the Manhattan LSTM model, we observe the Manhattan LSTM model only works with naive similarity comparisons without analyzing the deep level of latent semantics meaning. Directly calculate and compare the word embedding vector distance through siames structure model is not capable for the LSTM network to obtain thorough comprehension on common sense from the COPA data.

### 1.2. BERT

**Bidirectional Encoder Representations from Transformers** makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. In its vanilla form, Transformer includes two separate mechanisms an encoder that reads the text input and a decoder that produces a prediction for the task. BERT learning model(LM) has obtained new state-of-the-art results on eleven natural language processing tasks, including the GLUE and SuperGLUE benchmarks. Currently 5 of the top scores on the SuperGLUE leaderboard are using BERT based models.Our team proposes to use the opensource BERT pre-training language model as the basis of our implementation for the COPA task.
There are two steps to the BERT model: pre-training and ne-tuning. During pre-training, the model is trained on unlabeled data over different pre-training tasks. For ne-tuning, the BERT model is rst initialized with the pre-trained parameters, and all of the parameters are ne-tuned using labeled data from the downstream tasks.

- As stated before, the COPA task provides a training dataset of 400 examples and evaluation dataset of 100 examples

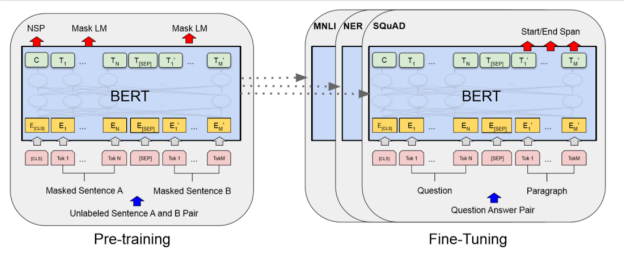


*Figure 2.* BERT

- We are generating new COPA training data by taking the MultiNLI training data, from the original GLUE task, and creating new COPA examples. The MultiNLI training set has over 10,000 examples.

- **MultiNLI to COPA:**

$$Entailment$$

$$COPA\ premise = initial\ sentence$$

$$COPA\ Alternative1 = entitled\ sentence$$

$$COPA\ Alternative2 = neutral\ sentence$$

$$COPA\ Choice = Alternative1$$

$$Neutral$$

$$COPA\ Alternative1 = entitled\ sentence$$

$$COPA\ Alternative2 = contradiction\ sentence$$

$$COPA\ Choice = Alternative1$$

$$Contradiction$$

$$COPA\ Alternative1 = contradiction\ sentence$$

$$COPA\ Alternative2 = neutral\ sentence$$

$$COPA\ Choice = Alternative2$$

- We will pre-process our data similar to how we would for the Siamese LSTM model, except this time we will add [CLS] to the start of the data field, and use [SEP] to separate the premise and the possible cause & effects.

$$Condition = cause$$

$$data1 = [CLS]query1[SEP]answer1$$

$$data2 = [CLS]query1[SEP]answer2$$

- BERT will create an contextual aware embedding vector for each word part in the two sentences. It will also create a embedding value for the tokens [CLS] and [SEP] which is based on sentence level embedding

- We then calculate the Euclidean distance between the embedded vectors for the word parts in the two sentences. We also calculate Cosine similarity between the [CLS] and [SEP] sentence level embedding.

- We would select the answer that minimizes these values.

## 2. Data exploration

Since Our initial BERT model got 64% training accuracy with batch size=5 and epoch = 10. However, the accuracy reduced after increasing the training epochs. After observing the data, we decide to attempt to classify the data with an additional label. The label allows us to separate or data into two sets called hard data and easy data. We did this by manually pre-processing the training data.

**Case 1:**
P: The man got a discount on his groceries.
A1: He greeted the cashier.
A2: He used a coupon
**Step 1:**Remove all stop words: the, a, on, his, he.
P: man, got, discount, groceries.
A1: greeted, cashier.
A2: used, coupon.
**Step 2:**Any pair words? **Yes, (DiscountCoupon)**
**Step 3:** Does the number of similar words $>= 50\%$? **No**
**Step 4:** Label 0 (Easy)

**Case 2:**
P: The heavyset man decided to lose weight.
A1: He cut out sweets.
A2: He avoided caffeine.
**Step 1:**Remove all stop words: the, to, out, he.
P: heavyset, man, decided, lose, weight.
A1: cut, sweets.
A2: avoided, caffeine.
**Step 2:**Any pair words? **No**
**Step 3:** Label 1 (Difficult)

**Case 3:**
P:I deposited the letter in the mailbox.
A1: The post office delivered the letter.
A2: The post office expedited the letter.
**Step 1:**Remove all stop words: I, the, in.
P: deposited, letter, mailbox.
A1: post, office, delivered, letter.
A2: post, office, expedited, letter.
**Step 2:**Any pair words?
**Yes,(mailboxletter)(mailboxpost, office)**
**Step 3:** Does the number of similar words $>= 50\%$? **Yes**
**Step 4:** Label 1 (Difficult)

**Algorithm 1** Sentence selection

**Input:** Premise $P_i$, Alternative1 $A_1$, Alternative2 $A_2$, Data Amount $N$
**repeat**
  Initialize $Label = [\,]$.
  **for** $i = 1$ **to** $N - 1$ **do**
    **if** $There\ are\ any\ pair\ words\ and\ Number\ of\ similar\ word > 50\%$ **then**
      $Label.append(1(Difficult))$
    **else if** $There\ are\ any\ pair\ words\ and\ Number\ of\ similar\ word < 50\%$ **then**
      $Label.append(0(Easy))$
    **else if** $There\ are\ not\ any\ pair\ words$ **then**
      $Label.append(1(Difficult))$
    **end if**
  **end for**
**until**

*Table 1.* Test accuracy on different type of training data

| MODEL | ACC(EASY DATA) | ACC(DIFFICULT DATA) |
|---|---|---|
| INITIAL BERT | 70% | 55% |
| INITIAL ROBERTA | 64% | 60% |
| SECOND ROBERTA | 67% | 58% |
| OUR BERT | 78% | 64% |

### 2.1. Model performance

After implementing data selection algorithm, we observe the model performance on each algorithm.

Submitted papers can be up to eight pages long, not including references, and up to twelve pages when references and acknowledgments are included. Acknowledgements should be limited to grants and people who contributed to the paper. Any submission that exceeds this page limit, or that diverges significantly from the specified format, will be rejected without review.

The text of the paper should be formatted in two columns, with an overall width of 6.75 inches, height of 9.0 inches, and 0.25 inches between the columns. The left margin should be 0.75 inches and the top margin 1.0 inch (2.54 cm). The right and bottom margins will depend on whether you print on US letter or A4 paper, but all final versions must be produced for US letter size.

The paper body should be set in 10 point type with a vertical spacing of 11 points. Please use Times typeface throughout the text.

## Acknowledgements

**Do not** include acknowledgements in the initial version of the paper submitted for blind review.

If a paper is accepted, the final camera-ready version can (and probably should) include acknowledgements. In this case, please place such acknowledgements in an unnumbered section at the end of the paper. Typically, this will include thanks to reviewers who gave useful comments, to colleagues who contributed to the ideas, and to funding agencies and corporate sponsors that provided financial support.

## References

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

## A. Do *not* have an appendix here

***Do not put content after the references.*** Put anything that you might normally include after the references in a separate supplementary file.

We recommend that you build supplementary material in a separate document. If you must create one PDF and cut it up, please be careful to use a tool that doesn't alter the margins, and that doesn't aggressively rewrite the PDF file. pdftk usually works fine.

**Please do not use Apple's preview to cut off supplementary material.** In previous years it has altered margins, and created headaches at the camera-ready stage.