Group 6: SemEval Task 4 – Commonsense Validation and Explanation

Utsav Jivani (1138453)
Beili Yin (1148875)
Parth Dhameliya (1132450)
Garima Chhaparwal (1131259)
Jaydeep Joshi (1160015)

Abstract- Natural Language Processing is an evolving task and we can understand and generate statement with the help of Natural Language Understanding (NLU) and Natural Language Generation (NLG). In this project, we portray our group's work in the Commonsense Validation and Explanation task, which is essential for SemEval2020. We assess five pre-processed Transformer-based language models (XLNET, ALBERT, BERT, DISTILBERT, ROBERTA) with different sizes against the three proposed subtasks. For the initial two subtasks, the best accuracy levels accomplished by our models are 86.50% and 89.90%. Concerning the last subtask, our models arrive at 0.9808 BLEU score and 1.94 human assessment score as per these two measurements, separately.

I. INTRODUCTION

A. Problem Statement

Create a model to test whether a system can differentiate statements that make sense or not. Several tasks that need to fulfil with the given criteria. First is to choose two statements with similar wordings, which one makes sense, and which one does not. Another task is to find a reason about the statement which is not making sense and final is to deliver a reason and evaluate them with BLEU [1].

Statement 1	John put a turkey into a fridge	Correct
Statement 2	John put an elephant into the fridge	Incorrect
	Elephant cannot fit into the fridge	Reason

Table 1: Problem statement with explanation

B. Explanation

NLP is a term in Machine Learning which gives the ability to a machine to generate, analyse, understand, and recreate human-like language. Although in past few decades the ability of a computer to understand common sense language has improved but it is still limited to a certain extent. Common sense reasoning plays a vital role in a Natural Language Understanding system and hence it is important to find out to what extent a computer can understand whether given statements make sense.

C. Motivation

In the recent years, NLP has evolved very rapidly. This field is further divided into many subdomains. Some of these subdomains are Natural Language Generation (NLG) and Natural Language Understanding (NLU). Common Sense Reasoning (CSR) is well-known application of NLU and NLG. A system that can commonly differentiate sentences, which makes sense or does not make sense. There has been lots of work done in the field of CSR. CSR is too hard problem to solve using any single artificial intelligence technique. In order to develop a common sense understanding capable system, one needs to incorporate multiple AI techniques. The human capacity to comprehend language is adaptable and powerful. Conversely, most NLU models over the word level are generated for a particular assignment and battle with out-of-area data.

II. RELATED WORK

A. NLU and CSR Based on LSTM, BERT and ngram models

Shrenik Atul Doshi [1] went a well-organized object on the overview, methodologies, experimental process and results, along with project discussion and conclusion. The Common-sense Validation and Reasoning task often consists of two parts: Natural Language Understanding (NLU) and Natural Language Generation (NLG). The purpose of the first part is to classify if a sentence makes sense in the context of real-world situation. To accomplish both tasks, two types of datasets were involved in this experiment. The dataset which consists of pairs of sentences that related to this task was manually labelled. Each pair has a reasonable sentence while the other one doesn't make sense. After per-processing steps the tokenized data was trained by BERT (base variant) and n gram. The sentences which ware classified as against common sense will serve as input for GPT-2 and Seq2Seq LSTM neural network for Reasoning Generation.

Limitation:
The proposed system is successfully meeting the needs of the problem statement, but the models developed for validation tasks did not outperform the existing pre-trained models in terms of accuracy. The output shows that the model was not trained adequately owing to a lack of training data and time limitations.

B. Common-sense Validation and Explanation Based on LMVE

Unlike Shrenik Atul Doshi's work, Liu, Shilei, et al [2] have done their research based LMVE, a proposed neural network model which includes two sub-modules to solve both sub-task (validation and reasoning) bases on large scale pre-training language model. For better performance the team adapted Pre-trained Language Models of ALBERT variants for training the data. This was where the team recognized their values and managed to incorporate them into the training models. To compile an Explaining Model, first starting with fine-tuning on Sen-Making Model, then adding encoder to extract information on the output of previous model and feed forward to the Explanation Generation Model. The existing Explaining Model was reversed to compile a Sense-Making Model. The process is parallel to the Generative Adversarial Neural Network. The final performances were decently plausible (at above 93 percent accuracy).

Limitations:
Because various language models employ different corpora, hyperparameters, and model structures during pre-training, these variances have an impact on the model's performance. To more completely represent each model's skill on this job, the authors did not incorporate new training data while fine-tuning, instead preserving the data supplied by the organiser.

III. METHODOLOGY

Task A: Validation (Sentence Classification)

We are using the same BERT model as we proposed earlier that perform sentence validation using BERT model and trying to generate the explanation sentence as we are facing the issues in the same. We also trying to run the model like GPT and DistlBERT to compare the different model.

The goal is to predict the sentence that contradicts common sense from a pair of sentences.
For Example:
● He put a turkey into the fridge.
● He put an elephant into the fridge. (Model predicts this as against)

The original data is formatted as follows: Id sentence0 sentence1
Labels are also in the following format: Id label
We divided each row into two sentences and assigned labels of 0 for incorrect and 1 for correct. As an example,

ID	Sentence	Labels
0	He wrote an exam in knife	0
0	He wrote an exam in pen	1
2	Chicken is found on a farm.	1
2	Chicken can swim in water.	0

Table 2: Subtask A Binary Classification example

We will have the labels in the following format

0	0
2	1

Table 3 : Label format

This indicates that sentence 0 in ID 0 is incorrect, and sentence 1 in ID 2 is incorrect. According to this, sentence0 is classified first, followed by sentence1. we did this for all of the data, including training, validation, and testing.

We will use BERT [3], GPT and DistelBERT [4] models to get Prediction. And finally, we only accept labels if they appear in all predicted labels or at least two of them.

OpenAI GPT:
The GPT model from OpenAI is a unidirectional pre-trained model that is based on the likelihood of the following word in a sequence. GPT is used to examine the perplexity scores. The sentence's confusion score indicates how this sentence does not make sense in certain aspects. The more perplexed you are, the less sense it makes.

Bert:
Bert is a pre-trained bidirectional model that employs bidirectional encoder representations. For bert, We utilised the 'bert-large-uncased' model. We will utilise the train and dev(val) datasets to fine-tune hyper-parameters and find the optimal model. We utilised a batch size of 32, 8 training epochs, a learning rate of 2e-5, and an eps value of 1e-8.

DistilBert:
We used 'distilbert-base-uncased' for DistilBert. DistilBert has a batch size of 8, four training epochs, and a learning rate of 5e-5.

Task B: Explanation (Multi-Choice)

The aim of sub task B is to predict which sentence from the given set is the most applicable reason to why a particular statement is against common sense. An example from the data set for task B is as follows

Statement S: He put an elephant into the fridge
A: An elephant is much bigger than a fridge.
B: Elephants are usually white while fridges are usually white.
C: An elephant cannot eat a fridge.

Option A is the anticipated value in the preceding case. As a result, the task is effectively a multi-class classification problem, with the target variables being option phrases A, B, and C. The models were trained to predict which of the three sentences, in terms of common sense, is the most favourable to the specific assertion S. The training input data was processed to separate each sentence and its three options into three records of data, so that each option sentence was concatenated along with their respective statements with a [EOL] tag between and an indication feature as class 1 or class 0 to indicate which sentence among the three is the expected answer. As a result, the data was processed to perform a binary text classification task, with class 1 indicating that the record contains the relevant reason why the sentence is against common sense and class 0 indicating that the record does not contain the relevant reason why the sentence is against common sense. After processing of input data for task B, different Transformer-based language models (XLNET, ALBERT, BERT, DISTILBERT, ROBERTA) were trained explicitly for the same task.

Model Architecture

The Neural Network models (XLNET, ALBERT, BERT, DISTILBERT, ROBERTA) were implemented using Hugging Face Library of Transformers (Wolf et al., 2019).

Bert:

Bert is a pre-trained bidirectional model that employs bidirectional encoder representations. For bert, We utilised the 'bert-large-uncased' model. We will utilise the train and dev(val) datasets to fine-tune hyper-parameters and find the optimal model. We utilised a batch size of 32, 4 training epochs, a learning rate of 2e-5, and an eps value of 1e-8.

DistilBert:

We used 'distilbert-base-uncased' for DistilBert. DistilBert has a batch size of 8, four training epochs, and a learning rate of 2e-5. We used 'distilbert-base-uncased' for DistilBert. DistilBert has a batch size of 8, four training epochs, and a learning rate of 2e-5. We used 'distilbert-base-uncased' for DistilBert. DistilBert has a batch size of 8, four training epochs, and a learning rate of 2e-5.

XLNet:

For XLNet, we utilised the 'xlnet-large-uncased' model. We will utilise the train and dev(val) datasets to fine-tune hyper-parameters and find the optimal model. We utilised a batch size of 32, 4 training epochs, a learning rate of 2e-5, and an eps value of 1e-8.

Alberta:

For Alberta, we utilised the ' albert-base-v2 ' model. We will utilise the train and dev(val) datasets to fine-tune hyper-parameters and find the optimal model. We utilised a batch size of 32, 4 training epochs, a learning rate of 2e-5, and an eps value of 1e-8.

RoBERTa:

For RoBERTa, we utilised the ' roberta-base' model. We will utilise the train and dev(val) datasets to fine-tune hyper-parameters and find the optimal model. In this model space will be treat as a separate token. so a word will be encoded differently whether it is at the beginning of the sentence (without space) or not We utilised a batch size of 32, 4 training epochs, a learning rate of 2e-5, and an eps value of 1e-8.

Post-Processing:

Individual model outputs were subjected to post-processing, in which the output probability scores of three phrases (records) for a certain statement are combined into a single record using high probability values for class 1. The processed outputs from various models were then combined into a final form of output using a majority vote technique.

Task C: Explanation (Text Generation)

We utilised the pre-trained GPT-1(Openai-GPT), GPT-2 and XLNET language model heads for subtask C and fine-tuned it using 3 referential reasons for each false statement as input (This task has nothing to do with Subtask B). The total number of false statements for training is 10000, which means 30000 training samples in total (Each referential reason is a concatenation of its corresponding false statement). (Code License: http://www.apache.org/licenses/LICENSE-2.0
Copyright 2018 The Google AI Language Team Authors and The HuggingFace Inc. team.
Copyright (c) 2018, NVIDIA CORPORATION. All rights reserved.)
GPT is known to train huge models with billions of parameters; for example, GPT-3's largest edition has 175B parameters. Its architecture is based on the Transformer's decoder block. The encoder-decoder cross attention part of the block is removed because there is no encoder, and the self-attention part is replaced with the masked self-attention.
GPT-2 is a large transformer-based language model with 1.5 billion parameters, trained on a dataset of 8 million web pages. GPT-2 is trained with a simple objective: predict the next word, given all of the previous words within some text. The diversity of the dataset causes this simple goal to contain naturally occurring demonstrations of many tasks across diverse domains. GPT-2 is a direct scale-up of GPT, with more than 10X the parameters and trained on more than 10X the amount of data.

XLNet is an auto-regressive language model which outputs the joint probability of a sequence of tokens based on the transformer architecture with recurrence. Its training objective calculates the probability of a word token conditioned on all permutations of word tokens in a sentence, as opposed to just those to the left or just those to the right of the target token. XLNet uses byte pair encoding of SentencePiece library which operates on Unicode strings and it was pre-trained on multiple datasets which amount to 136 GB of data.

The GPT-1, GPT-2, XLNET language model head are transformers with a language modelling head on top (linear layer with weights tied to the input embeddings) specifically for test generation.

All 3 models were fine-tuned under the same hyperparameters with 32 batch size, 5 Epochs and 128 Block size.

IV. EXPERIMENTS AND RESULTS

For subtask A and B, we evaluate our result with accuracy and confusing matrix. For subtask C, the performance can be evaluated with BLUE score. For now, we have not proceeded to subtask B and subtask C. There are 10,000 pairs of sentences available for training, 997 pairs of sentences available for validation [1], and 1000 pairs of sentences available for testing.

Task A: Validation (Sentence Classification) As shown in the table

Models	**Learning Rate**	**Batch size**	**Epochs**	**Accuracy**
BERT	2e-5	32	8	**83.99%**
GPT	-	-	-	**75%**
DISTILBERT	5e-5	8	4	**85.6%**
OVERALL	-	-	-	**86.5%**

Table 4: Validation Experimental Results

As shown in the Table 4: Task A: Validation (Sentence Classification) Experimental Results A comparison between BERT’s large-uncased' model, DISTBERT and GPT is made it is observed that DISTIBERT outperforms the other two models Our submission is based on majority voting ensemble between three models. This ensemble performs better than any other model on the development set. The Overall accuracy of ensemble is 86.5%

Task B: Explanation (Multiple Choice)

Comparing RoBERTa base results with remaining four models results shows that it is better than all other model for this problem statement. Our submission is based on majority voting ensemble between different models. This ensemble has different accuracy in different combinations the highest accuracy is obtained by combining all models (Xlnet + Albert + Bert + Distilbert + Roberta) while the lowest accuracy is obtained in this (Bert + Distilbert + Roberta) combination.

Models	Learning Rate	Batch Size	Epochs	Accuracy
Bert	2e-5	32	4	71.06%
Distilbert	2e-5	32	4	71.96%
Xlnet	2e-5	4	4	68.17%
Roberta	2e-5	32	4	77.76%
Albert	2e-5	32	8	72.76%

Table 5: Explanation (Multi-Choice) Single Model Experimental Results

Merged Models	**Accuracy**
Xlnet + Albert + Bert + Distilbert + Roberta	89.9%
Albert + Bert + Distilbert + Roberta	88.1%
Bert + Distilbert + Roberta	85.3%
Albert + Distilbert + Roberta	86.7%
Albert + Bert + Roberta	87.1%

Table 6: Explanation (Multi-Choice) Merged Model Experimental Results

Task C: Explanation (Text Generation)

For evaluation, the predicted reasons are assessed against 3 referential reasons. the metric for evaluation is BLEU, short for Bilingual Evaluation Understudy, which is a score for comparing a candidate translation of text to one or more reference translations. Although developed for translation, it can be used to evaluate text generated for a suite of natural language processing tasks. a BLEU implementor is to compare n-grams of the candidate with the n-grams of the reference translation and count the number of matches. These matches are position independent. Although BLEU is an official tool to assess the experimental results of subtask C, it doesn’t map well to human judgements. For example, if the generated reasons are well justified in human common sense but do not match with the 3 referential reasons, the BLEU score will be brought down. Aside from the major flaw of BLEU,

Model	**BLEU Score**
GPT-2	0.9808
OPENAI-GPT	0.8679
XLNET	1.2267

Table 7: Explanation (Text Generation) Experimental Results

Conclusion

As part of the Common-sense Validation and Explanation task, we assessed pre-trained Transformer-based language models against three common sense tasks. Our research demonstrated that pre-trained language models may be used to extract and validate facts as strong knowledge bases. We find that training a Transformer model to take a single phrase as input and identify the sentence within the pair separately can get good results in subtasks A and B.

References:

[1] “SemEval-2020 Task 4: Common-sense Validation and Explanation”, Wang, Cunxiang and Liang, Shuai long and Jin, Yili and Wang, Yilong and Zhu, Xiaodan and Zhang, Yue ,”Proceedings of The 14th International Workshop on Semantic Evaluation”, ”Association for Computational Linguistics”, ”2020”

[2] “Common sense validation and reasoning using Natural Language Processing”, Shrenik Doshi, Praveen Joshi, Haithem Afli, ”2020”.

[3] Url : https://huggingface.co/ [online]

[4] Url : https://swatimeena989.medium.com/bert-text-classification-using-keras-903671e0207d [online]

[5] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation.

Workload Distribution:

- **Utsav Jivani:**
  Worked on Subtask B using Roberta and XLNET language model.
- **Garima Chhaparwal:**
  Worked on Subtask A using BERT, DISTILBERT language model.
- **Parth Dhameliya:**
  Worked on Subtask B using BERT, DISTILBERT language mdoel.
- **Beilli yin:**
  Worked on Preprocessing task in Subtask B and used GPT-2 to perform Subtask C and calculated BLEU score.
- **Jaydeep Joshi:**
  Worked on Subtask A using GPT and did research related work.