Myers-Briggs Personality Classification on Reddit Data using BERT with Abstractive Summarization

Richard Du
UC Berkeley School of Information
richard.du@berkeley.edu

Abstract

The Myers-Brigg Type Indicator is a widespread and pervasive personality test that relies on four distinct categories that define a person's personality. Recently, the use of pre-trained models such as BERT have been introduced to attempt to classify people's personality based on labeled texts written by those people. However, BERT has an inherent limitation with its max input sequence length of 512 tokens. In this paper, we investigate whether using a Longformer model on a novel Reddit dataset to summarize texts before being input to BERT can increase classification performance. The baseline model without summarization reached a peak accuracy of 0.339 and F1 score of 0.335 and the summarized dataset produced worse results with an accuracy of 0.212 and F1 score of 0.201. However, the task of personality classification is still important and perhaps future new summarization techniques will help overcome these challenges.

Introduction

The Myers-Briggs Type Indicator (MBTI) is a commonly used personality test developed in 1995 by Katharine Briggs and Isabel Myers (Myers et al. 1995) and was based on the conceptual theory formulated by Carl Jung. These types of personality tests are commonly used by the public as well as in companies, with 89 of fortune 100 companies utilizing the tests either for hiring or within the workplace (Nguyen et al. 2018).

The MBTI test is composed of four distinct dichotomies comprised of attitudes (extraversion (E) / introversion (I)), functions (sensing (S) / intuition (I) and thinking (T) / feeling (F)) and lifestyle preferences (judging (J) / perception (P)). The MBTI test classifies an individual as a combination of one selection from each dichotomy, which results in 16 distinct personalities. For example, a person could be classified as ESFJ, with one letter from each dichotomy.

Personality classification using online forums, more specifically with social media, is an ongoing area of curiosity for the natural language processing and social science worlds (Nguyen et al. 2016). Currently, one popular approach to personality classification of the MBTI system is using pre-trained language models such as BERT to classify personalities (Keh & Cheng 2019). In the paper, the authors used user comments to classify participants into an MBTI personality type and were able to achieve an accuracy of 0.47 in predicting all four dichotomies and an accuracy of 0.86 for predicting at least two of the four dichotomies. However, with BERT there are inherent limitation to max sequence length, in part due to the self-attention operation, which is a quadratically scaling function with the sequence length. This puts a max sequence length of 512 tokens as input to BERT. Due to the sequence length limitations, the Keh paper simply truncated inputs that were too long. One possible solution would be to use abstractive summarization techniques to reduce the input sequence length. They hypothesis is that information that may be lost by simple truncation may be maintained via abstractive summarization. This project seeks to improve MBTI personality classification performance.

Goal/Objective

This project will seek to utilize a similar methodology to Keh to replicate their results in a different dataset collected from a different source. Another goal of the project will be to select an abstractive summarization technique to improve overall model performance.

The main metric of evaluating success in this project will be to achieve a classification accuracy significantly improved over randomly guessing in addition to comparing the model performance to similar papers that attempt to classify MBTI personality types based on text inputs (Keh & Cheng 2019, Plank & Hovey 2015, Gjurkovic & Snajder, 2018, Cui & Qi, 2018). The other metric will be comparing the performance of models fine-tuned on the original dataset to models fine-tuned on the summarized dataset. These two metrics will allow to achieve the objective of attempting to replicate MBTI personality classification in an alternate dataset and the objective of improving classification performance on MBTI personality types.

Background

Personality Prediction Modalities

MBTI personality types have recently begun to be utilized for personality predictions. One of the first papers predicting MBTI personalities used logistic regression on a Twitter dataset (Plank & Hovy, 2015). Another paper using SVM and a different Reddit dataset followed (Gjurkovic & Snajder, 2018). A paper using LSTM's was also published using a Kaggle Dataset from Personality Café (Cui & Qi, 2018). Lastly, a BERT-based classifier was published using a separate Personality Café dataset (Keh & Cheng 2019).

Pre-Trained Language Representation Models

BERT is a pre-trained deep bidirectional transformer that has drastically changed the NLP landscape (Devlin et al. 2018). BERT pre-trains bidirectional representations from text via joint conditioning on left and right context, thus any pre-trained BERT model allows for fine-tuning by adding an additional output layer. BERT exemplifies transfer learning and can provide excellent results on a variety of tasks via fine-tuning the initial pre-trained model, including classification and question answering. Yet, the application of BERT to personality classification datasets has only recently been published. The idea of what to do with user texts that extend past the 512-sequence length limit of BERT has yet to be explored.

Longformer is a long-document transformer that addresses the problem of quadratic scaling with sequence length that plagues BERT via an attention mechanism based on multiple stacked layers of fixed-size attention windows (Beltagy et al. 2020). This combined with global attention results in a well-performing model that scales linearly with sequence length. Longformer architecture also contains a variant that includes both the encoder and decoder Transformer stacks that also allows for application in abstractive summarization with up to 16 thousand input tokens.

Methods

Dataset

There are no standardized datasets containing MBTI personality types and associated text. The data was scraped using Google Big Query on July 30th 2018 from Reddit on all subreddits that contained a reference to an MBTI personality type (Storey 2018). The dataset is similar in nature to previous papers that used data from Personality Café, Reddit and Twitter (Keh & Cheng 2019, Plank & Hovey 2015, Gjurkovic & Snajder, 2018, Cui & Qi, 2018). The dataset contains Reddit flairs, which are self-assigned tags that can contain MBTI

personality types, user comments and subreddit name. The original dataset contained 1,794,016 comments. The data was then divided into an 85-15 train-test ratio

Data Cleaning and Preprocessing

First, it was necessary to isolate the author flairs, as many from the original dataset contained aberrant tags. The tags were filtered to identify the user's MBTI personality type. Next, since the objective of the paper was to use long strings of text, only posts that contained 500 or more characters were kept in the dataset and shorter posts may not contain any meaningful information. Furthermore, due to computational constraints and imbalanced class sizes, 500 example texts of each personality type or the max number of example texts in personality types with fewer than 500 examples were selected. The final resulting dataset contained 6,468 Reddit comments. Class distribution can be seen in Table 1.

MBTI Personality	N
ENFJ	500
ENFP	500
ENTJ	500
ENTP	500
ESFJ	79
ESFP	214
ESTJ	82
ESTP	335
INFJ	500
INFP	500
INTJ	500
INTP	500
ISFJ	487
ISFP	271
ISTJ	500
ISTP	500

Table 1: Number of comments per personality type

Classification and Abstractive Summarization Methodology

Tokenization

Following with Keh & Cheng's paper, BERT's custom tokenizer from BERT-base-uncased was implemented. Standard truncation when sequence length exceeded the specified max sequence length and padding was used to fill shorter input sequences. Special tokens were added to the front and end of sentences, "[CLS]" and "[SEP]" respectively.

BERT Fine-Tuning and Training

Similar to the Keh & Cheng paper, the BERT model created was selected from HuggingFace's BERTforSequenceClassification, which is the BERT transformer paired with a regression head on top. Training was implemented based on techniques found in a guide (Li 2020). Training was implemented via fine-tuning the pre-trained BERT-base model on the training data. Training was done with batch size of 3. The Adam optimizer was selected and cross entropy loss was used as the loss function.

Longformer2RoBERTa Architecture

A Longformer2RoBERTa architecture was used for abstractive summarization. The encoder segment is Longformer model and the decoder is a RoBERTa-base model. Reddit posts were tokenized using the Longformer Tokenizer, longformer-base-4096 model. The model was fine-tuned by HuggingFace user patrickvonplaten on the CNN/Daily mail dataset for 90 hours first.

Results and Discussion

Tuning Hyperparameters

The three hyperparameters that were optimized were learning rate, max sequence length and number of epochs. Furthermore, in addition to looking at accuracy as a performance metric, due to the class imbalance, F1 score was also evaluated as a model performance metric.

Learning Rate	Max Seq.	Epochs	F1 Score	Acc.
.0001	128	20	0.011	0.077
.0001	256	5	0.011	0.077
.00001	128	30	0.307	0.302
.00001	**256**	**5**	**0.335**	**0.339**
.00001	256	30	0.331	0.322
.00001	512	5	0.331	0.339
.000001	256	10	0.225	0.243

Table 2: Baseline BERT model results

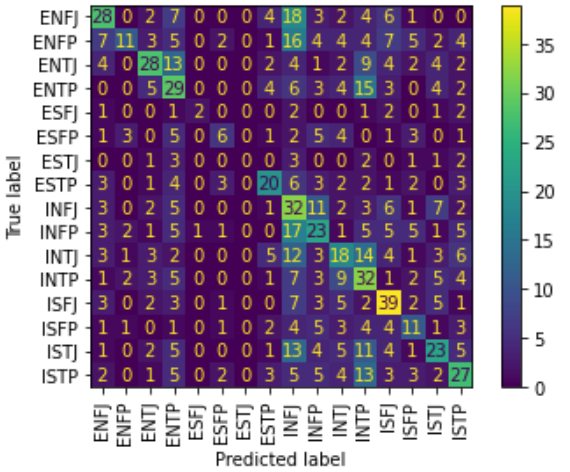

Figure 1: Confusion matrix of baseline model with the best F1 score.

Another common metric to analyze model performance used by other papers was to calculate the total number of correctly predicted letter in the MBTI score (Cui & Qi 2017). For example, if the model predicted ESFJ to the true label of ESTP, the model still predicted 2 of the 4 dichotomies correctly.

At least 1 match	At least 2 matches	At least 3 matches	At least 4 matches
0.975	0.867	0.616	0.339

Table 3: Number of correctly predicted MBTI dichotomies of baseline model.

The last metric to analyze model performance is to evaluate overall accuracy on a single dichotomy.

E/I	N/S	F/T	P/J
0.689	0.670	0.720	0.636

Table 4: MBTI dichotomy prediction accuracy of baseline BERT model

Longformer2RoBERTa

After summarizing the posts using the Longformer2RoBERTa architecture, the average post character length decreased from 1070 characters to 263 characters, approximately a 75% decrease in user post character length.

Summarized Dataset Hyperparameter Tuning

Learning Rate	Max Seq.	Epochs	F1 Score	Acc.
.0001	256	30	0.011	0.077
.00001	256	30	0.202	0.199
.00001	512	5	**0.201**	**0.212**
.000001	256	30	0.124	0.133

Table 5: BERT model trained on summarized user posts.

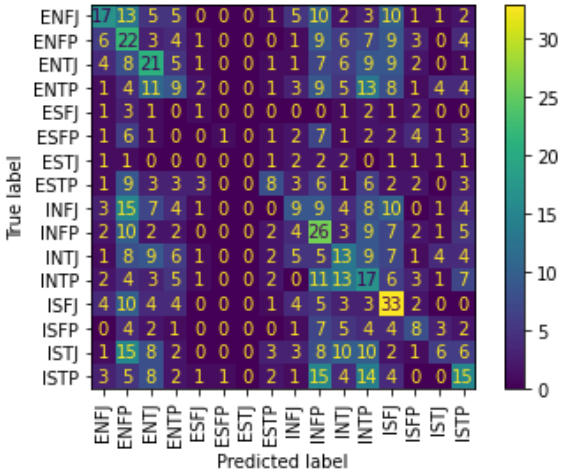

Figure 2: Confusion matrix of BERT mode trained on summarized data with the best performance.

At least 1 match	At least 2 matches	At least 3 matches	At least 4 matches
0.954	0.793	0.494	0.212

Table 6: Number of correctly predicted MBTI dichotomies of BERT model trained on summarized data.

E/I	N/S	F/T	P/J

0.599	0.613	0.641	0.600

Table 7: MBTI dichotomy prediction accuracy of BERT model trained on summarized data.

Discussion

From analysis of the baseline BERT model, a few key ideas become apparent (Table 2.). The baseline BERT model with the highest overall performance on this dataset had the following hyperparameters: learning rate = 0.00001, epochs = 5 and max sequence length = 256. This model had an F1 score of 0.335 and an accuracy of 0.339, which indicates it is performing much better than random. Furthermore, the table gives insight towards the effect of each hyperparameter on model performance. Learning rate appeared to have the biggest effect on model performance with learning rate equal to 10^-4 having the worst performance of 7.7% accuracy which is barely better than randomly guessing. Max sequence length also appeared to have slight effects on model performance as max sequence increased, model performance also tended to increase. For example, increased max sequence length from 128 to 256 tokens at a learning rate of 10^-5 led to an increase of 2% accuracy. Interestingly, number of epochs did not appear to have a significant impact on model performance When looking at the training and validation loss across epochs, the validation loss stopped improving after approximately 20 epochs and the validation classification performance plateaued after about the 5^{th} epoch. Meanwhile, these finding slightly differ from Keh & Cheng's paper. Keh & Cheng's paper had a peak accuracy of 0.4797 compared to our accuracy of 0.339, meaning the model was underperforming. Their paper also found a significant effect of learning rate on model performance; however, they did not see an effect of max sequence length on their model performance and did see an effect of number of epochs on model performance. One potential explanation could be the difference in datasets, the dataset in this paper was specifically selected to include longer user posts, which could explain the effect of sequence length on model performance. Furthermore, the difference in the size of the datasets could be an explanation since their dataset was 10x the size. One other explanation is the sensitivity of BERT to class imbalance. The significant class imbalance (Table 1) may have affected our model training and overall performance. From these findings, the hyperparameter testing on the BERT model trained on the summarized dataset could be further optimized based on these results.

Looking at the baseline confusion matrix (Figure 1.), a few trends appear. The model seems to have trouble differentiating extraverts from introverts. The model classified nearly 25% of the ENFJ personalities as INFJs with 18 of the 75 labels getting the E/I dichotomy incorrect. Furthermore, distinguishing specifically between INFP's and INFJ's seems particularly difficult for the model with 15% of the INFJ's being classified incorrectly as an INFP and 23% of INFP's being incorrectly classified as an INFJ.

Moving onto Table 3., the model successfully predicts at least 1 dichotomy correctly 97.5 of the time and this number decreases as the number of dichotomies increases. The overall accuracy scores and closely follows the published numbers using BERT and, in some cases, even slightly outperforms the published results (Keh & Cheng 2019). Keh & Cheng's model had at least two dichotomies correct 85.7% of the time and this model was correct 86.7% of the time.

Next, looking how the model performs individually on each dichotomy (Table 4.), the model is the best at distinguishing F from T with an accuracy of 0.720 and E from I with an accuracy of 0.689. These results are similar to those found in Keh & Cheng's paper. Their model also performed the best on those two dichotomies. This could potentially indicate that these personality traits are particularly sensitive to language and language structure.

After using the abstractive summarization algorithm, the average character length of a post was shortened from 1070 to 263, a 75% decrease. However, model performance also decreased (Table 5.). Similar trends appear in terms of which hyperparameters influence the model performance the most. Learning rate still has the biggest impact. The model with the best performance had a learning rate = 10^-5, max sequence length of 512 and 5 epochs with a F1 score of 0.201 and accuracy of 0.212. Yet, this is still a marked decrease compared to the baseline model. Closer inspection of the confusion matrix illustrates an overabundance of predicting ENFP for other MBTI types compared to the baseline model. Furthermore, performance on correctly predicting at least two dichotomies decreased going from an accuracy of 0.867 to 0.793. From Table 7, there is a difference in the relative performance of differentiated certain dichotomies. The two dichotomies that had the most relative performance loss as a result of the abstractive summarization were E versus I and F versus T, with a 13% and 11% decrease in accuracy, respectively. One potential reason for this may be that abstractive summarization excels at maintaining the salient ideas of the source text, rather than maintaining the text's writing style necessarily. Previous research has shown that a person's writing style rather than simply the ideas expressed in text are related to personality structure (Ruffner & Burgoon 1981). This may be one explanation for the decrease in model performance as the writing style associated with the personality type was lost as a result of summarization.

Conclusion and Future Directions

Based on the results of this paper, we successfully replicated the BERT-based personality classification model presented in Keh & Cheng's paper in a separate and smaller dataset comprised of long-form user comments from Reddit with an accuracy of 0.339 and a F1 score of 0.335. The model performance was approximately 29% worse compared to their paper. Furthermore, using the Longformer model for abstractive summarization to attempt to minimize the effect of BERT's 512 token input sequence limitation yielded approximately 30% performance degradation compared to baseline.

Due to time constraints and computational limitations, certain tradeoffs were made. In the future with fewer time and computational limitations, using a larger BERT model such as BERT-Large will be investigated. Furthermore, one possibility to improve performance would be to increase the dataset size to 20,000 entries and include more of the shorter input texts into the dataset. Another area of future work would be to try other abstractive summarization algorithms that may maintain more of the source text's style.

References

Brandon Cui and Calvin Qi. 2017. Survey analysis of machine learning methods for natural language processing for mbti personality type prediction.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. CoRR, abs/1810.04805.

Matej Gjurkovic and Jan Šnajder. 2018. Reddit: A gold mine for personality prediction. NAACL HLT 2018, page 87.

Sedrick Scott Keh & I-Tsun Cheng. (2019). Myers-Briggs Personality Classification and Personality-Specific Language Generation Using Pre-trained Language Models.

Susan Li (2020). Multi class Text Classification with Deep Learning Using BERT. https://towardsdatascience.com/multi-class-text-classification-with-deep-learning-using-bert-b59ca2f5c613

Myers, Isabel Briggs with Peter B. Myers (1995) [1980]. Gifts Differing: Understanding Personality Type

Dong Nguyen, A. Seza Dog˘ruöz, Carolyn P. Rosé, and Franciska de Jong. 2016. Computational socio-olinguistics: A survey. Computational Linguistics 42(3):537–593.

Janet Nguyen. How companies use the Myers-Briggs system to evaluate employees (2018). https://www.marketplace.org/2018/10/30/myers-briggs-system-evaluate-employees/

Barbara Plank and Dirk Hovy. 2015. Personality traits on twitterâA˘T˘orâA˘T˘how to get 1,500 personality tests in a week. pages 92–98, 01.

Ruffner M, Burgoon M. The Relationship between Writing Style and Personality Structure. Newspaper Research Journal. 1981;2(2):28-35. doi:10.1177/073953298100200203

Dylan Storey. (2018). Myers Briggs Personality Tags on Reddit Data (0.0.1) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.1323873