# An Emotion Classification Scheme for English Text Using Natural Language Processing

Mose Gu, Junhee Kwon, and Jaehoon (Paul) Jeong
*Department of Computer Science and Engineering*
*Sungkyunkwan University*
Suwon, Republic of Korea
{rna0415, juun9714, pauljeong}@skku.edu

Sanghee Kwon
*Department of Media and Communication*
*Sungkyunkwan University*
Suwon, Republic of Korea
skweon@skku.edu

***Abstract*—With the development of Natural Language Processing (NLP), Artificial Intelligence (AI) has reached the level of understanding the context of long and complex sentences and interpreting the meaning. Since the AI model pre-trained with a large amount of data can produce high classification performance through a little fine-tuning, we aim to fine-tune and evaluate the two state-of-the-art pre-trained models with a dataset consisting of rich emotions to classify the various emotions. In order to show the potential, we evaluated two state-of-the-art models such as Bert [1] and Electra [2], and compared their performance in emotion classification.**

***Index Terms*—AI, Deep Learning, NLP, Sentiment Analysis**

## I. Introduction

Natural Language Processing (NLP) with deep learning has been developed continously based on transformer and has shown a remarkable capability in understanding context and generating translation sequence. Among the transformer-based NLP models, a pre-trained encoder language model called BERT, which has emerged as a representative model in NLP in particular and also has been honored as the state-of-the-art NLP model for a long time. Furthermore, it has become a baseline for many in pre-trained language model researches. With the advance in the language model, it is possible to carry out many different NLP tasks such as text generation and question and answering. Sentiment analysis, which is one of the representative tasks of NLP, is a task that classifies the sentiment of a given sentence, and is an important evaluation index in analyzing customer participation in companies.

Due to the development of sentiment analysis, the importance of emotion analysis has also emerged day by day on many platforms (e.g., a news portal site), due to the fact that emotion analysis is a field that deeply understands and uses people's opinions, attitudes, and tendencies. Unlike sentiment analysis, emotion analysis judges polarity to classify the emotion into joy, sadness, anger, and so on against an object, situation, or atmosphere. Therefore, it is safe to say that emotion analysis has the advantage of constructing a more sophisticated analysis of emotions from the text than sentiment analysis. However, existing sentiment analysis studies only determine positive, negative, and neutral mood in context. Thus, the sentiment analysis has a limitation that only performs simple classification disregarding the variety of emotions.

According to this necessity, we propose a framework to supplement the limitation of the simplicity of sentiment analysis. We conducted our framework to evaluate deep learning-based emotion classification with standard emotion words (e.g., joy, embarrassment, anger, anxiety, sadness). For the experiment, we have compared the performance of two different state-of-the-art models such as BERT and ELECTRA which are already pre-trained with the Wiki dataset, through fine-tuning and evaluating with rich emotion literature datasets in order to figure out a better model for emotion classification.

The main contributions of this paper are summarized as follows.

- **Survey and Evaluation of the State-of-the-Art Models:** We studied and surveyed State of the art models and compared them by analyzing classification performance for specific tasks. We have contributed to finding out which models perform better in various classifications of emotions throughout our framework.
- **Emotion Classification with Literature Dataset:** It was necessary to look forward to a dataset that is rich in emotion representation for various emotion classifications. Using data collected from drama scripts, novels, and poems and labeled with the help of a liberal arts department, we contributed to the classification power of the model's reading comprehension ability and emotional understanding.
- **A Framework for Model Comparison:** We used a confusion matrix of f1 scores to compare and analyze the performance of the two State of the art models. Through this, we could contribute to evaluating and comparing which models are particularly strong in classification and understanding, and which classes are classified well.

The remainder of this paper is organized as follows. The background and survey of our research are described in Section II. Section III describes the overview of the proposed architecture and explains the components and implementation. Section IV shows the detailed performance evaluation with the experiment results of the models. Section V concludes this paper.

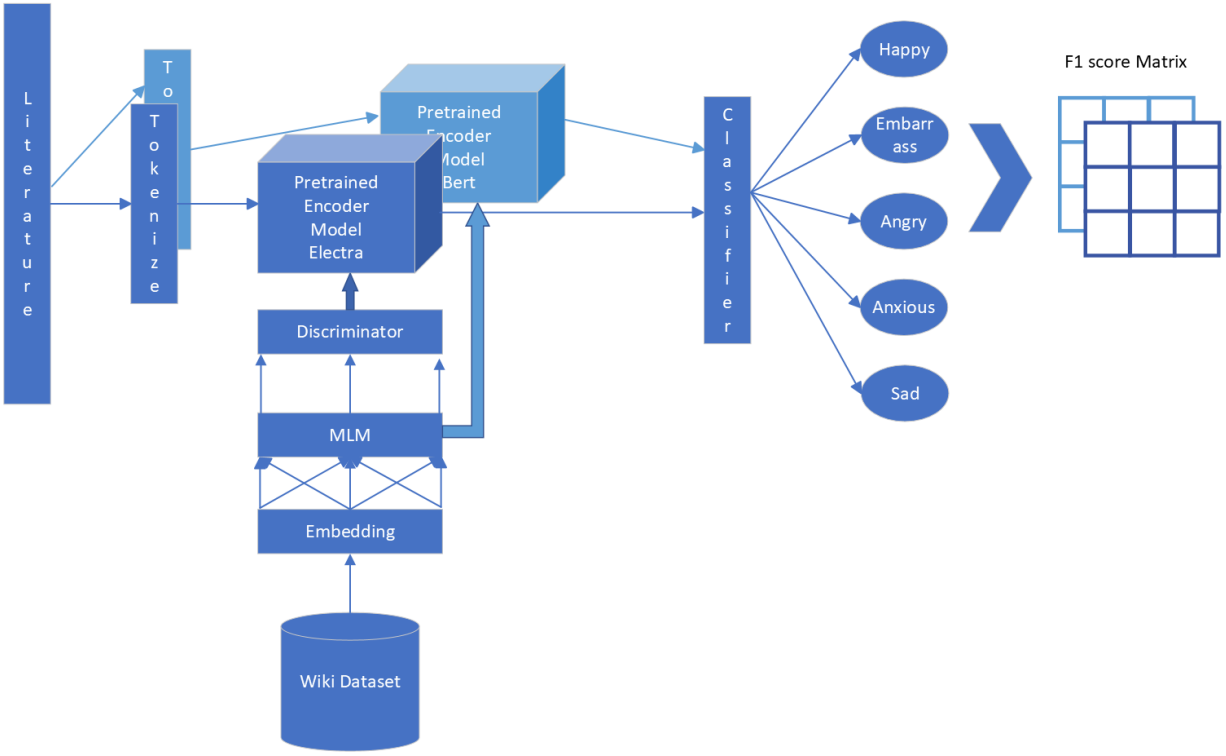



Fig. 1: An emotion classification comparison framework for SOTA pre-trained encoder models

## II. Related Work

### A. Sentiment Analysis

Sentiment Analysis is being studied as a constant GLUE score task in the field of deep learning. Bert said that at the time of publication, 11 GLUE scores produced the state-of-the-art performance [3], and the GLUE score is the criterion for determining the state-of-the-art one. Likewise, for the NLP model to be nominated as a state-of-the-art model, it must be demonstrated as an NLU model generalized through several NLP tasks. Sentiment analysis is one of the representative studies in these GLUE. Several companies, as well as GLUE, contribute to sentimental analysis research by providing sentimental analysis datasets (e.g., IMDB) for commercialization.

### B. BERT

The Transformer proposed by Vaswani et al. [4] was an effective structure to extract the attention of the input sequence in order to overcome the long-term dependency that LSTM's overwriting the information up to the existing timestep at each timestep [5], but a simple combination of Encoder and Decoder allowed room for further improvement. Google's Bidirectional Encoder Represents from Transformers (i.e., BERT) is a structure organized to accumulate layers of the original Transformer Encoder structure to fully reflect the correlation between words.

Due to the auto-regressive architecture of the original language model, all input tokens can only attend to the previous token, which limits the power of representation [1]. However, by implementing the Masked Language Model (i.e., MLM) that randomly masks a certain token of whole input tokens within a sentence in order to induce the model to predict the masked tokens, this pre-training task enabled self-attention in both directions which increased power in contextual representation with showing excellent performance in predicting the next word [1].

This representation also reflected better performance in fine-tuning. Because the aforementioned BERT had a very distinct performance improvement over the existing LSTM techniques, in the NLP field, BERT replaced the existing methodology as the state-of-the-art NLP model.

### C. Electra

Kevin et al. [2] announced the language model, Effectively Learning an Encoder that Classifications Token Replacements Accurately (i.e., ELECTRA), which applied the new pre-training technique. Many State of the art language models, including existing BERT, pre-train through the masked language modeling task, which replaces the input with a mask token and restores it to the original token before the replacement. However, these models mask and restore 15 percent of the tokens in the input sequence when learning, so it takes a significant amount of computation to lose only 15 percent of all tokens. Therefore, because learning is expensive, ELECTRA is paying attention to the accuracy of the model and the efficiency of learning. To improve learning efficiency, the author proposes a new pre-training task called Replaced

Token Detection (i.e., RTD), which allows ELECTRA to learn faster and more effectively.

The RTD task is implemented with two encoder network, a generator and a discriminator, similar to the GAN [6] algorithm. The discriminator learns the token sequence generated by the MLM network Generator in binary classification whether each token is original or replaced. While MLM only learns 15 percent of sample tokens, the RTD task allows Electra to learn 100 percent of tokens through a discriminator network.

As a result, ELECTRA outperformed traditional BERTs under the same conditions of model size, data, and computing resources.

## III. Design and Implementation

### A. Architecture for SOTA Comparison

The overall architecture of our scheme is shown in Fig. 1. This architecture gives a general description of our scheme. Components are broadly composed of five components: A literature dataset consisting of 5 classes for fine-tuning, a tokenizer that is an embedding layer that transforms data into a trainable datatype, two SOTA models, a classifier that classifies the 5 output logit through softmax, and F1 confusion matrix that shows the performance and evaluation of the model.

Literature data is split into multiple tokens through a tokenizer and input to the model. At this time, the models receiving the input are models whose parameters have been initialized after training has already been completed. The input examples are outputted as representations after each token operation is performed through the model and input together with the class in the classifier layer. The input example that has passed through the classifier layer is output as a 5-dimensional vector, and the final value with the highest probability through the softmax layer is subjected to supervised learning by calculating the cost according to the same class as the original input. When error backpropagation is performed through supervised learning, the model fine-tunes numerous pre-trained parameters to the input data, used during training.

### B. Components for Implementation

This subsection describes the component and implementation

**Literature:** Literature is a literature dataset for fine-tuning for our task (emotion classification). The dataset has been collected from three literary data (e.g., drama script, novel, poetry) with maximized polarity in proportion to analyze objective forms of quantified information from subjective data such as people's opinions, attitudes, and tendencies. As shown in Fig. 2, data with maximized polarity from the data collected from comedies, novels, and poetry scripts are made up of basic human emotions (e.g., happy, embarrassed, angry, anxious, heartbreaking, sad). To avoid imbalances in data in which data from a particular class appear at very high frequencies, we prepared a dataset of similar proportions for each label. In Fig. 2, the distribution of data by class can be confirmed. The variance of each data is less than 20 by the mean of 650.

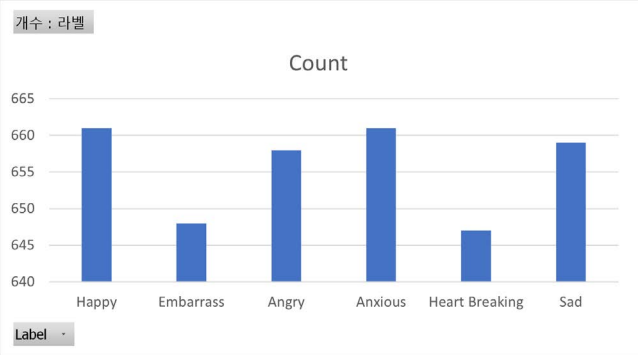



Fig. 2: Classifier for feature extracted representations

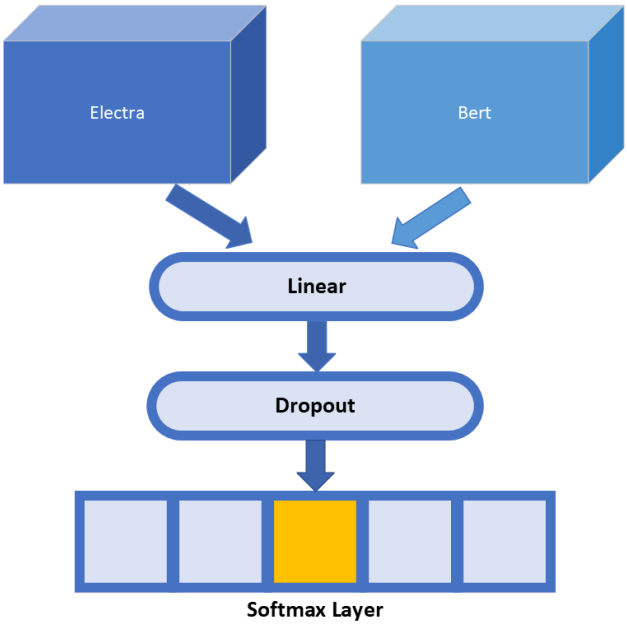
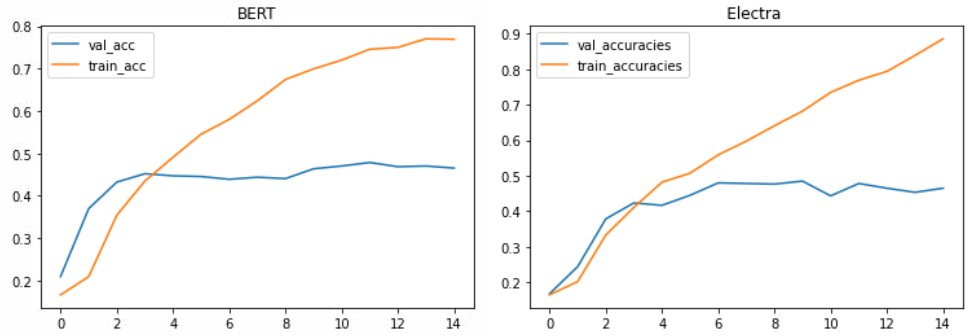



Fig. 3: Classifier for feature extracted representations

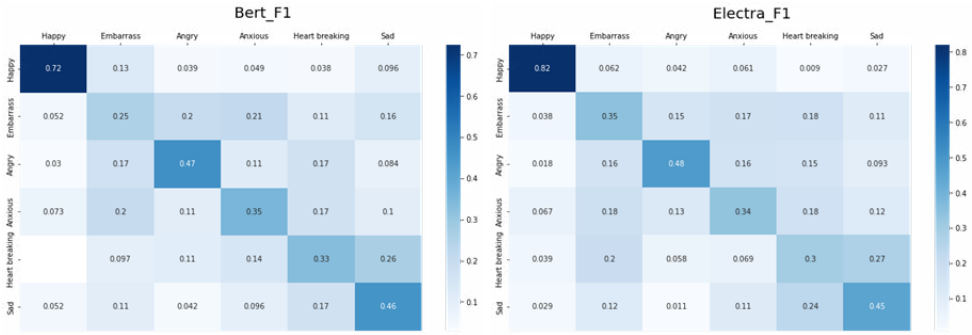
**Tokenizer:** Tokenizer is a preprocessing task to enable the model to understand our literary data. TThe tokenizer used wordpiece embeddings to split the input sequence into subwords. The word piece embedding merges the corpus together based on the frequency and the highest pairs of likelihoods to produce the subwords that accord to the context [7]. The input example passed through the tokenizer is converted into the subwords tokens of the specified length, and the remaining space is filled with padding. After Tokenizing, the subword has contextual information, helps the model understand the context of the input example. When the input example passed through the tokenizer, it is set to a sequence length of 512 and trained with a maximum of 6 batch sizes. While the iteration, 6 input examples perform computation in parallel within the model.

**Pre-trained Model:** Pre-trained model is a model that is already trained and parameters are initialized with the trained datasets. The benefit of the pre-trained model is clear. The way to build a model with good performance is to acquire




(a) Classification accuracy in training and validation for Bert

(b) Classification accuracy in training and validation for Electra




(c) F1 score matrix of Bert

(d) F1 score matrix of Electra

Fig. 4: Evaluation metric for 6 classes

a large number of data, but since it is expensive, a method to use a part of a neural network trained in a specific field for training a neural network used in a new field is proposed to solve this limitation [8]. The trained neural network used for transfer learning is called a pre-trained model. The pre-trained models we selected are BERT, and Electra. The two models were completed learning with a massive wiki dataset. We chose the state-of-the-art models such as Bert and Electra among the bidirectional learned encoder models. Both models have learned the representation of the Wiki Dataset with the different learning tasks. Bert has a representation extracted through the MLM technique, and Electra, another model, has a representation extracted through the RTD technique. Our framework aims to evaluate the classification performance of RTD and MLM with the same dataset. Bert and Electra both understand context and meaning well, hence it can be expected to measure the understanding power to the emotional expression of the literary data set.

**Classifier:** Classifier is a transfer learning layer for the pre-rained model. The pre-trained models are convenient for transfer learning of various tasks because they extract representations from large amounts of inexpensive, unlabeled data. Since the pre-trained model is a huge feature representation, it is possible to use it for transfer learning by implementing a classification layer suitable for the task. Fig. 3 shows a detailed structure of the classifier hierarchy. The classifier outputs the logit of the class with the highest probability through softmax among the number of outputs equal to the number of classes. The output through the classifier is implemented to output one of five classes through a linear layer and a dropout layer.

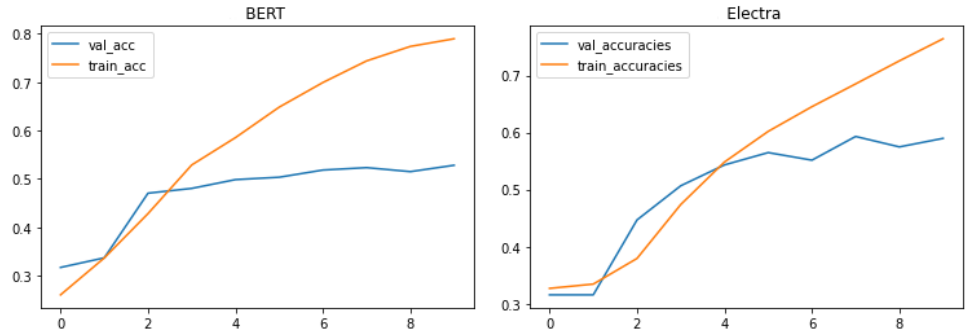
**F1 Score Matrix:** The equation shown in equation 1 is the F1 score that has been selected for the evaluation metric. We used a confusion matrix for the metric visualization. The confusion matrix is an evaluation index for checking how well the model predicted for each class by class. Our confusion matrix is filled with an F1 score for absolute evaluation under any circumstances. Detailed confusion matrix output values can be found in section 4. The F1 score was selected as an evaluation metric because it can prove that it is a good model in any situation as an evaluation index that considers objectives and various situations by reflecting both precision and recall.

$$F1score = 2(Precision * Recall)/Precision + Recall \quad (1)$$

## IV. Performance Evaluation

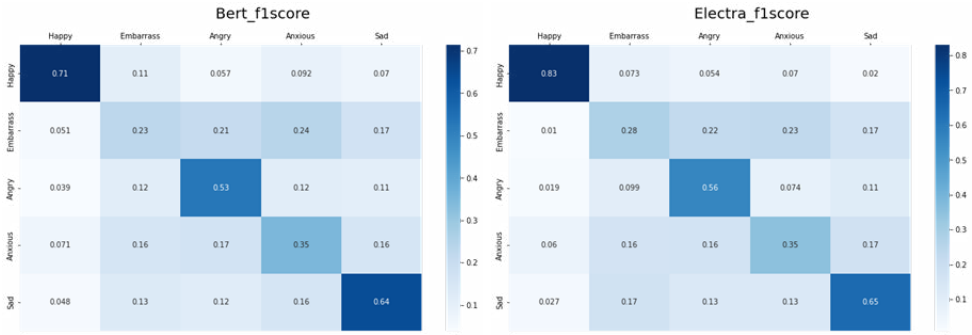
This section contains a summary of the performance of emotion classification. The experiment proceeded by outputting the model accuracy and f1 score matrix with data samples of the same size. Denote that hyperparameters (e.g., learning rate, optimizer, loss function) are all the same to reduce the error between models.

The result shown in Fig. 4 is the fine-tuning classification accuracy using 6 classes of emotions in the training and validation stages of the model. The first experiment, which started in the black box environment, was conducted with 15




(a) Classification accuracy in training and validation for Bert

(b) Classification accuracy in training and validation for Electra




(c) F1 score matrix of Bert

(d) F1 score matrix of Electra

Fig. 5: Evaluation metric for 5 classes by merging two similar classes into one class

epochs, the point where the training accuracy peaked. As a result of observation, Bert's training convergence speed was faster than Electra in the same sampling, but the Electra model had a slower learning speed compared to the Bert model, while the accuracy was higher than that of the relatively stable Bert model. In addition, in the Electra model, overfitting was observed at the 10th epoch point. This shows that the RTD-based Electra model has higher model complexity than the MLM-based BERT model.

Fig. 4 shows that Electra is a more sophisticated model than Bert, and as proof, the F1 score matrix shows that Electra was higher than Bert's f1 score. Neither model showed dramatic accuracy, but the purpose of our study, 'Comparing the performance of two models on the same sample', showed significant results. In particular, Electra recorded Happy class 0.1 points higher than Bert. It means Electra has better model power to distinguish positive samples. Other than that, the results were similar for both models, but we captured a spot with a widely distributed heatmap in the f1 score matrix; Heartbreaking class and sad class.

Those classes are having high similarities, so it is safe to say that the confusion between the two classes was high enough to confuse two given models. Reflecting on these black-box results, we decided to observe the results of the 10th epoch before the over-fitting phenomenon occurred in the next experiment. Also, after merging the data of two classes with high similarity, heartbreaking, and sad classes, the experiment was conducted by re-sampling with the same number of other classes.

The results shown in Fig. 5 are the results of subsequent experiments. As a result of observation, based on the training accuracy of the Bert model, the convergence speed of Bert was faster than that of Electra. However, as a result of checking the validation accuracy, Bert showed stable results, while Electra's continuous performance increased at the utmost 60 percent accuracy was confirmed. On top of this, by merging the two classes with high similarity, the understanding of the input sample of Electra was further increased, and it can be confirmed that the predicted values for all classes of Electra in the f1 score have increased than Bert.

## V. Conclusion

This section concludes our research. We compared and analyzed the model power of how much the two State of the art models understood the emotional context. Through the experiment, we have demonstrated that the RTD pre-training task is superior to the MLM pre-training task in emotion classification. From the viewpoint of the existing Sentiment analysis research, it was confirmed through the f1 score of Happy, that Electra was far superior to Bert in the classification performance of negative and positive. Although the classes other than the happy class, which are positive in emotion classification, are related to each other and have high similarities, so a perfect classification can be difficult. Nevertheless, the performance increase through the

RTD technique has been clearly confirmed, and it appears that there is ample room for improvement.

## ACKNOWLEDGMENTS

This work was supported by the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center (ITRC) support program (IITP-2022-2017-0-01633) supervised by the Institute for Information & Communications Technology Planning & Evaluation (IITP). This work was supported in part by the IITP grant funded by the Korea MSIT (No. 2022-0-01199, Regional strategic industry convergence security core talent training business). Note that Jaehoon (Paul) Jeong is the corresponding author.

## REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018. [Online]. Available: https://arxiv.org/abs/1810.04805

[2] K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, "Electra: Pre-training text encoders as discriminators rather than generators," 2020. [Online]. Available: https://arxiv.org/abs/2003.10555

[3] Z.-X. Ye, Q. Chen, W. Wang, and Z.-H. Ling, "Align, mask and select: A simple method for incorporating commonsense knowledge into language representation models," 2019. [Online]. Available: https://arxiv.org/abs/1908.06725

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017. [Online]. Available: https://arxiv.org/abs/1706.03762

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[6] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014. [Online]. Available: https://arxiv.org/abs/1406.2661

[7] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean, "Google's neural machine translation system: Bridging the gap between human and machine translation," 2016. [Online]. Available: https://arxiv.org/abs/1609.08144

[8] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural Computation*, vol. 29, no. 9, pp. 2352–2449, 2017.