Webology, Volume 18, Number 4, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I4/144

Implementation of Traditional Vs. Transformer Machine Learning Models

Noor Mohd[1], Gushan Dhasmana[1], Deepak Upadhyay[2]

[1]Department of Computer Science & Engineering Graphic Era Deemed to be University,
Dehradun, Uttarakhand, India
[2]Department of Computer Science & Engineering, Graphic Era Hill University, Dehradun,
Uttarakhand India, 248002

ABSTRACT

Classical models have been mostly used in NLP tasks. There are several classical models like Naïve Bayes, SVM, CNN etc. In classical model we need to build the model from scratch. The model is trained, developed and then lastly deployed. With the advent of transformer models the process have been simplified. Transformer models are pretrained on larger datasets and just needs labelled dataset. It saves time, energy and resources. This report discusses the mechanism involved in workflow of classical and transformer ml models. Implementation of models have been also done for comparison.

Keywords: ML Models, NLP, Transformers Model, AI.

INTRODUCTION

Machine learning

Man-made intelligence is a piece of man-made Artificial Intelligence (AI) and programming that bright lights on supporting assessments that can look like or outperform human understanding. The learning measure is persistent and bases on extending precision. It is a huge progression portion in the field of information science. Classical ML methods, profound learning, and neural organizations are a portion of the classes for AI models. Multiclass classification is discussed in [6, 7].

Classical Machine learning

The methodology are used in this, what start with the component assurance, extraction, and coming about getting ready. Fostering the model beginning from the soonest stage a long time. There are three essential procedures for learning:

Supervised Learning: The use of a model to encourage ability with an organizing between input models and the target variable is suggested as directed learning. Models are fitted to data contained information sources and yields and used to make assumptions on test sets where basically the information sources are given and the model's yields are stood apart from the guaranteed target sections to consider the model's capacity.

Gathering, which requires predicting a class name, and backslide, which consolidates expecting a numerical worth, are the two fundamental sorts of controlled learning endeavors.

- Classification: Supervised learning issue that incorporates anticipating a class mark.
- Regression: Supervised learning issue that incorporates anticipating a numerical imprint.

Somewhere around one data components can be used in both request and backslide issues, and data elements can be of any data type, for instance, numerical or straight out.

The MNIST genuinely made digits dataset, where the data sources are pictures of deciphered digits (pixel data) and the yield is a class mark for which digit the image addresses, is a framework of a depiction issue (numbers 0 to 9).

The Boston home expenses dataset, where the data sources are factors that portray a region and the yield is a house cost in dollars, is an outline of a backslide issue. Since they are proposed for oversaw AI issues, certain AI computations are implied as "managed" AI estimations. Decision trees, support vector machines, and significantly more models are ordinary.

Unsupervised Learning: Unsupervised learning refers to a class of problems where correlations in data are explained or eliminated using a model. Unlike supervised learning, unaided learning only affects the input data and does not take yields or goal parameters into account. In contrast to controlled learning, independent learning avoids the need for an instructor to alter the model. Although there are several forms of unaided learning, there are two common issues that professionals must deal with: clustering, which entails recognizing groups in the data, and thickness appraisal, which combines summarizing the dispersed material.

- Clustering: Unsupervised learning issue that incorporates finding packs in data.
- Density Estimation: Unsupervised learning issue that incorporates summarizing the scattering of data.

K-Means is a delineation of a social occasion estimation, where k implies the amount of groups to arrange in the data. Part Density Estimation is an outline of a thickness examination calculation that wires using little groupings of immovably related information tests to survey the legitimacy for new fixations in the issue space. To get some answers concerning the models in the data, bunching and thickness evaluation can be used. Unaided techniques like acumen, which incorporates charting or showing data in a sudden manner, and projection philosophy, which join diminishing the dimensionality of the data, may similarly be used. Representation: Unsupervised learning issue that fuses making plots of information. Projection: Unsupervised learning issue that consolidates making lower-dimensional portrayals of information. A dissipate plot framework, for example, causes one to disperse plot for each pair of parts in the dataset. Head Component Analysis is a representation of a projection strategy that fuses summarizing a dataset to the degree eigenvalues and eigenvectors, with straight conditions disposed of.

Reinforcement learning: Support learning suggests a group of problems where a professional operates in a certain setting and must determine how to do so after careful consideration. The utilization of an environment suggests that there should not be a predefined planned dataset, but rather a goal or group of goals that a professional should reach, possible actions they could take, and analysis of their progress toward the goal. 3 It is comparable to coordinated learning in that the model has some response to learn from, despite the fact that the information may be delayed and quantifiably chaotic, making it difficult for the educated expert or model to link circumstances with reasonable outcomes.

Playing a game where the expert's point is to get a high score and may make advancements in the game and get contribution as disciplines or rewards is an outline of a support learning.

Q-learning, fleeting contrast learning, and profound support learning are some conspicuous support learning calculations.

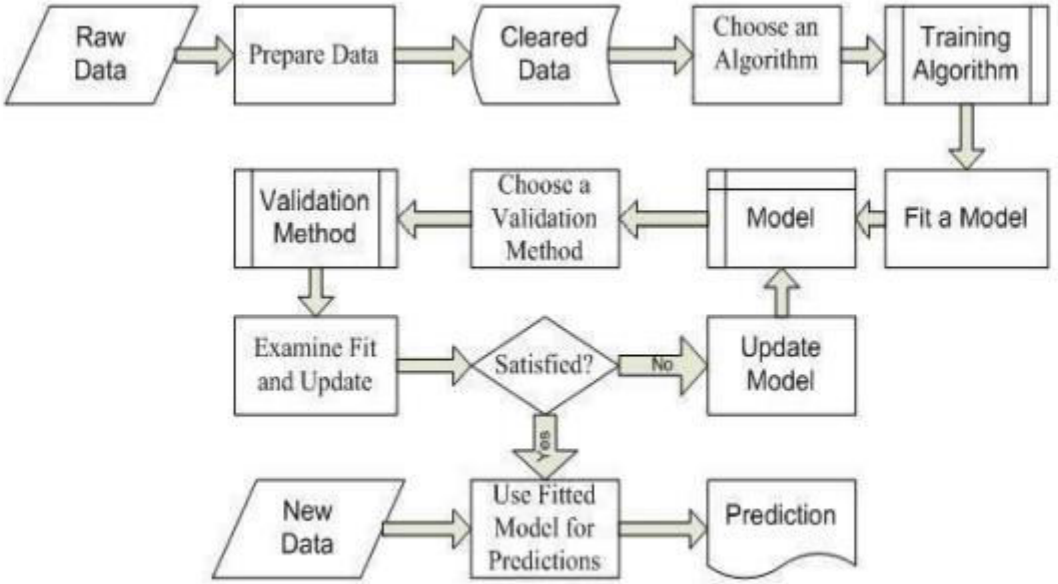


Figure 1: Basic Machine learning workflow diagram

transformer ML model

Transformers are a well-known example of neural association planning. Transformers were actually utilized by Open AI in their language models and by Deep Mind for Alpha Star, their weapon of choice for defeating a top-tier StarCraft player. Transformers were developed to address the problem of data transduction, also known as neural machine interpretation. Any venture that alters an information course of action as a result is combined into a yield gathering. This combines text-to-speak transition, talk affirmation, and other features. The Transformer in NLP is a new arrangement that handles short-term planning efforts while easily managing long-term situations. In the paper Attention Is All You Need, the Transformer was suggested. Anyone who is enthusiastic about NLP should investigate.

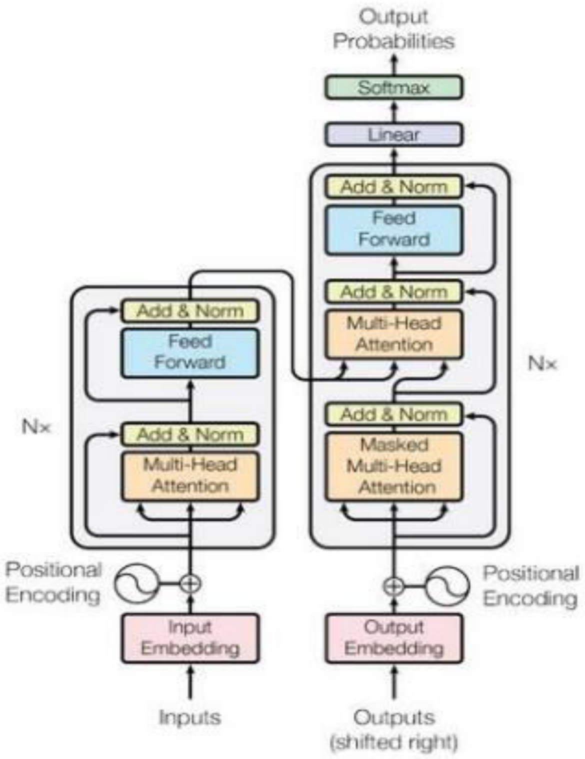


Figure 2: Transformer Architecture

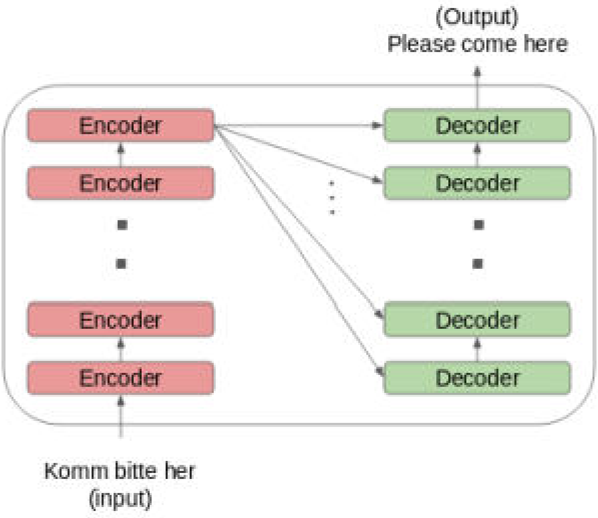


Figure 3: Encoder and decoder

We ought to see how this outline of the encoder and the decoder stack works:

- The word embeddings of the information gathering are given to the first encoder
- These are then changed and scattered to the going with encoder
- The yield from the last encoder in the encoder-stack is passed to the sum of the decoders in

the decoder-stack.

Something significant for see here is that, in spite of oneself idea and feed-forward layers, the decoders contain one extra layer of Encoder-Decoder Attention. This permits the decoder to zero in on the significant pieces of the data movement.

Limitations of Transformers

Transformer is unquestionably an enormous improvement over RNN-based seq2seq models. It does, regardless, go with countless impediments:

- Only fixed-length text strings can be used to focus thought. Before being managed in the framework as data, the material ought to be apportioned into a foreordained number of portions or parts.
- Text clumping encourages a fractured setting. For instance, much of the setting is lost if a sentence is split down the middle. The content is then isolated without regard for the articulation or any other semantic limitations.

BERT

BERT is a model for natural language processing. BERT stands for bidirectional encoder representations from transformers. In 2018, Google Research researchers made the suggestion. It is built on transformers, a deep learning model where every input and output element is coupled. Based on their connections, the weighting between the input and output parts is dynamically determined. "Attention" is the Natural Language Processing term for this action. In terms of handling NLP and NLU tasks, it set numerous records. The model's code was also made available for download after they had described it. This pre-trained model conserves the effort, information, time, and resources required to train a model from scratch. The classifier must be trained primarily in order to train this model, with the BERT model undergoing the fewest modifications possible. Fine tuning is a training method with roots in ULMF and semi-supervised sequence learning. Therefore, in order to train such a model, we require a labelled dataset.

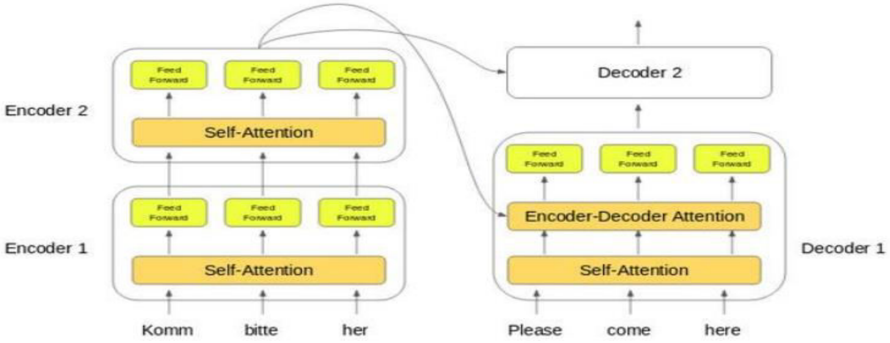


Figure: 5 BERT's Model Architecture

BERT's Model Architecture

A multi-layer bidirectional Transformer encoder is used by BERT. Its care layer performs care in the two different ways. Google has conveyed two variations of the model:

1. BERT Base: Total limits = 110M, number of transformer layers = 12.
2. BERT Large: 24 transformer layers, full scale limits = 340M

BERT uses bidirectionality by coordinating two endeavors: Masked Language Model and Next Sentence Prediction.

BERT's Pre training task
BERT is pre-customized with the assistance of the two unaided expectation errands recorded underneath.

Masked Language modeling:
The Google AI experts covered 15% of the words in each gathering at abstract. The objective was to expect the verbalizations that have been covered. An outpouring of alert: considering the way that the secretive tokens [MASK] would never be seen during transforming, they were not by and large used to uproot the covered words. Therefore, the specialists utilized the going with procedure:

- 80% of the time, the mysterious token [MASK] was utilized to substitute the words.
- The words were subbed with abstract ones 10% of the time.
- The words were left unaltered 10% of the time.

Next Sentence prediction:
The link between the subsequent enunciations is consistently overlooked by language models. BERT was also prepared for this task. BERT creates language models by using explanation sets to gather data. It's interesting to see how each pair of sentences is chosen. Expect to need to pre-train a BERT language model using a dataset of 100,000 sentences from a book. There will therefore be 50,000 getting ready models or sentence blends as planning information.

- The next sentence would really come after the important sentence in half of the sets.
- The supplementary sentence would be an out-of-the-ordinary sentence from the corpus for the additional section of the sets.
- The titles for the significant occurrence would be "IsNext" and "NotNext" for the succeeding case.

Models Unsupervised learning (status and changing) will be a vital part in different language learning frameworks, as displayed by models like BERT. These critical bidirectional models, expressly, can give colossal advantages to low resource endeavors.

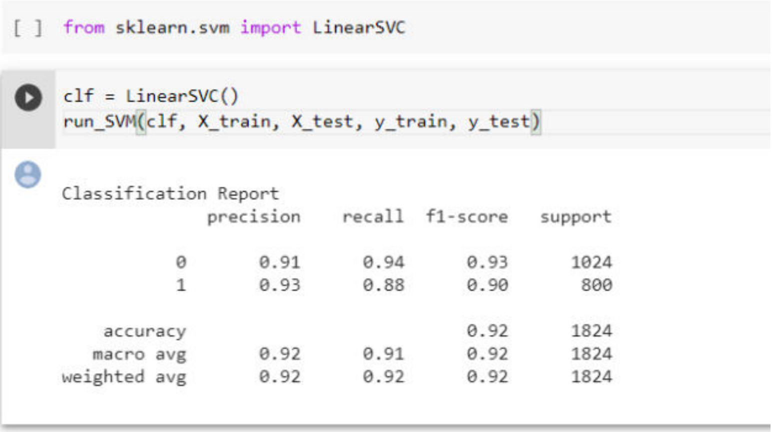
Implementation
In order to show difference between traditional machine learning model and Transformer model both models have been compared by the way of implementation. For traditional I implemented binary classification using TfIdf and SVM, Word2Vec and SVM, etc. For Transformer model I implemented BERT Model. First we need to clean the data. The noise can be in any form from blank spaces, urls, emojis to hashtags etc. Text cleaning and pre-processing module of NLP is used in this process. Text cleaning also helps to increase the efficiency of the module then we loaded our cleaned dataset and performed ML techniques to classify the texts. In our framework we took

Webology, Volume 18, Number 4, 2021
ISSN: 1735-188X
DOI: 10.29121/WEB/V18I4/144

genuine posts of social media where people have freedom to express their views briefly. Dataset was chosen from https://www.kaggle.com/nikhileswarkomati/suicidewatch. It includes posts that were made to "Suicide Watch" between December 16, 2008 (when it was created) and January 2, 2021, whereas posts about depression are from January 1, 2009, to January 2, 2021. I cleaned the data using NLP techniques before applying the necessary models. Testing and comparing the model that has been shown in the next part comes next.

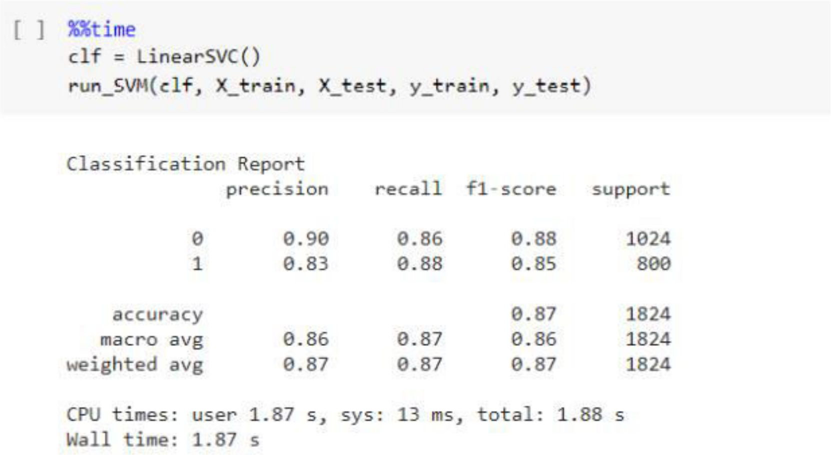
Result

In classification using tf-idf and SVM the accuracy is 92% ,in classification using word2vec and SVM the accuracy is got 87 % and in BERT model it’s 98.57.The difference between traditional and transformer model can be seen in output and it is huge in terms of accuracy and time constraint. In terms of binary classification traditional model is doing good but in multilabel classification it will lag behind from transformer models like FastText and Roberta



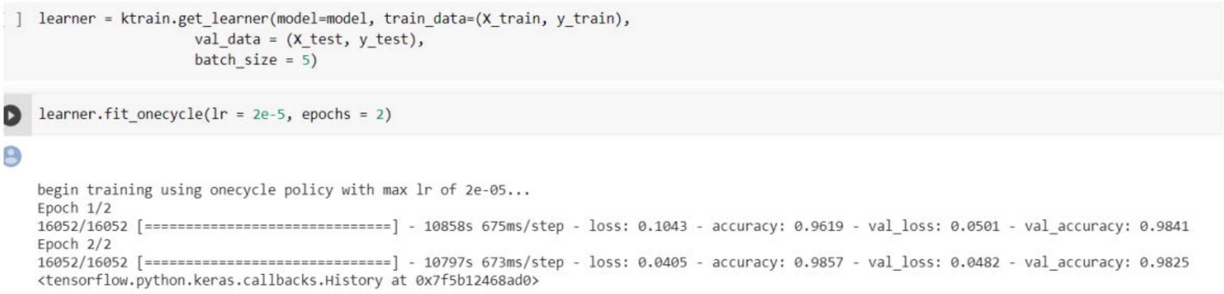
```
[ ] from sklearn.svm import LinearSVC

clf = LinearSVC()
run_SVM(clf, X_train, X_test, y_train, y_test)

Classification Report
              precision    recall  f1-score   support

           0       0.91      0.94      0.93      1024
           1       0.93      0.88      0.90       800

    accuracy                           0.92      1824
   macro avg       0.92      0.91      0.92      1824
weighted avg       0.92      0.92      0.92      1824
```

Figure 6: Classification with TFIDF and SVM



```
[ ] %%time
    clf = LinearSVC()
    run_SVM(clf, X_train, X_test, y_train, y_test)

Classification Report
              precision    recall  f1-score   support

           0       0.90      0.86      0.88      1024
           1       0.83      0.88      0.85       800

    accuracy                           0.87      1824
   macro avg       0.86      0.87      0.86      1824
weighted avg       0.87      0.87      0.87      1824

CPU times: user 1.87 s, sys: 13 ms, total: 1.88 s
Wall time: 1.87 s
```

Figure 7: Classification with `Word2Vec` and SVM



```
learner = ktrain.get_learner(model=model, train_data=(X_train, y_train),
                    val_data = (X_test, y_test),
                    batch_size = 5)
```

```
learner.fit_onecycle(lr = 2e-5, epochs = 2)
```

```
begin training using onecycle policy with max lr of 2e-05...
Epoch 1/2
16052/16052 [==============================] - 10858s 675ms/step - loss: 0.1043 - accuracy: 0.9619 - val_loss: 0.0501 - val_accuracy: 0.9841
Epoch 2/2
16052/16052 [==============================] - 10797s 673ms/step - loss: 0.0405 - accuracy: 0.9857 - val_loss: 0.0482 - val_accuracy: 0.9825
<tensorflow.python.keras.callbacks.History at 0x7f5b12468ad0>
```

Figure 8: Classification Using BERT Model

conclusion

There are other transformer models also like XLNET,ROBERTA,FASTTEXT,etc which show much more state of art accuracy in every segment from image classification, text filtering, classification, etc.The traditional models take too much time to implement from scratch and it requires technical expertise too. Though transformer models takes a good amout of gpu to implement but it is easy to use and is more efficient too. We should be glad since so many advanced developments occur at a very quick speed in NLP. Models like Transformers and BERT will make a much-developed leap ahead in the years to come.

References

1. Kaggle.com. 2021. r/SuicideWatch and r/depression posts from Reddit. [online] Available at: [Accessed 10 April 2021].
2. S. García, J. Luengo and F. Herrera, "Data Sets and Proper Statistical Analysis of Data Mining Techniques", 2021.
3. Devlin, M. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", arXiv.org, 2021. [Online]. Available: https://arxiv.org/abs/1810.04805?source=post_page. [Accessed: 21- Jun- 2021].
4. GitHub. 2019. google-research/bert. [online] Available at: [Accessed 19 April 2021]. 29. A. Dai and Q. Le, "Semi-supervised Sequence Learning", arXiv.org, 2021. [Online]. Available: https://arxiv.org/abs/1511.01432. [Accessed: 22- Jun- 2021].
5. A. Joulin, E. Grave, P. Bojanowski and T. Mikolov, "Bag of Tricks for Efficient Text Classification", arXiv.org, 2016.
6. Tiwari, P., Upadhyay, D., Pant, B., & Mohd, N. (2022). Multiclass Classification of Disease Using CNN and SVM of Medical Imaging. In *International Conference on Advances in Computing and Data Sciences* (pp. 88-99). Springer, Cham.
7. Tiwari, P., Pant, B., Elarabawy, M. M., Abd-Elnaby, M., Mohd, N., Dhiman, G., & Sharma, S. (2022). CNN Based Multiclass Brain Tumor Detection Using Medical Imaging. *Computational Intelligence and Neuroscience*, *2022*.