Investigating the Use of Statistical Tests in Machine Learning Models

Jule Valendo Halim -1425567
University of Melbourne

Abstract

Objective

To investigate possible roles of statistical tests on natural language processing tasks through models for multi-label and multi-class classifiers.

Materials

Dataset containing online patient reviews of prescribed medication, provided by Zoolnori et al[1] and Zoolnori et al[2]. The dataset contains annotated labels for each review sentence, along with the medication being reviewed.

Methods

Multi-label classification was performed with two different textual features, and each feature was trained on a logistic regression and transformer model. This resulted in 4 models.

Multi-class classification was performed with three different textual features, and each feature was also trained on logistic regression and transformer models, resulting in 6 models.

Model prediction were then evaluated using traditional machine learning metrics such as accuracy, before then being evaluated using statistical tests.

Results

When comparing traditional machine learning metrics, transformer models generally outperform their logistic regression, except for one task-feature pair. Statistical test results were reported and interpretations were provided.

Discussion

Statistical result interpretations were discussed as possible model behaviours that could be further investigated. The difficulty of applying statistical tests on natural language processing tasks were also discussed.

Conclusion

Statistical tests provide valuable insight into model behaviour outside of traditional machine learning metrics. However, additional work is needed in creating a pipeline to investigate causes of possible model behaviour, as well as integration of assumption testing.

Introduction

Statistical tests are a popular and long-standing method in multiple fields of science. However, studies in natural language processing (NLP) and machine learning (ML) do not generally include statistcal testing as part of their model evaluation. A survey on 233 published papers in the field of NLP showed that 132 of these papers did not report statistical significance[3]. However, more studies have begun advocating for the use of these tests to show that experimental results are not coincidental[3] and argue that a combination of NLP and statistical tests can provide a framework for the development of robust, high-throughput health NLP systems[4].

In this report, I aim to investigate the use of statistical tests on ML models that predict multi-label and multi-class classification tasks using the statistical test workflow suggested by Rainio, Tauho, and Klén[5], as seen in figure 1. However, instead of comparing results from different test sets, this report compares different model feature inputs.

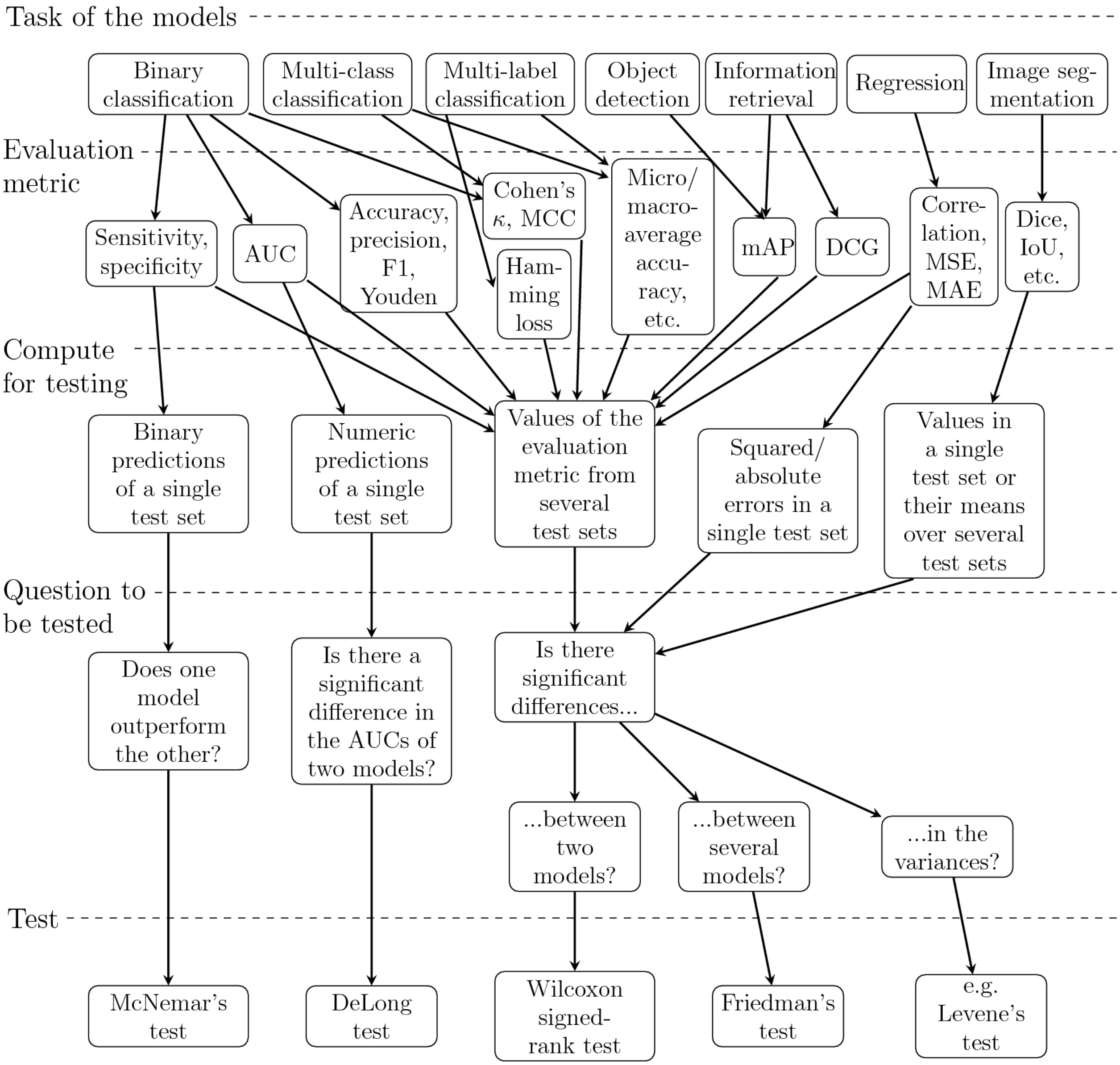
Task of the models
Binary classification
Multi-class classification
Multi-label classification
Object detection
Information retrieval
Regression
Image segmentation
Evaluation metric
Sensitivity, specificity
AUC
Accuracy, precision, F1, Youden
Cohen's κ, MCC
Hamming loss
Micro/ macro-average accuracy, etc.
mAP
DCG
Correlation, MSE, MAE
Dice, IoU, etc.
Compute for testing
Binary predictions of a single test set
Numeric predictions of a single test set
Values of the evaluation metric from several test sets
Squared/ absolute errors in a single test set
Values in a single test set or their means over several test sets
Question to be tested
Does one model outperform the other?
Is there a significant difference in the AUCs of two models?
Is there significant differences...
...between two models?
...between several models?
...in the variances?
Test
McNemar's test
DeLong test
Wilcoxon signed-rank test
Friedman's test
e.g. Levene's test

Figure 1. Statistical Workflow Provided by Rainio, Tauho, and Klén[5]

The Sentence_Labeling sheet of the PsyTAR dataset used in this study is provided by Zoolnoori et al[1] and Zoolnori et al[2]. This sheet contains three sections of interest. The first is sentences from online patient reviews of certain drugs. The second is annotations of certain labels, described below in table 1. The third is drug labels, which indicates which drug each sentence is reviewing.

Two classification tasks were performed using a baseline logistic regression model and a transformer model. The multi label class predicts six binary annotations, while the multi-class predicts four different drugs. For each model, different features will be used as inputs. The resulting predictions will be used for statistical tests in order to determine their statistical signifance. Specific details are described in the methods section.

Methods

Ethics Statement

This study has one important ethical consideration, which is that

Preprocessing

Two main preprocessing steps were done. Firstly, as the drug labels in the data were concatenated with review ID, the review ID was stripped and only the drug label was added to a new column. Secondly, the review text was preprocessed through tokenization, as well as stopword and punctuation removal. The dataset consists of 6009 records. The data was split into train, validation, and test sets using a 45:45:10 ratio.

Multi-Label Classification

Multi-label classification involves predicting six labels, as shown in table 1. Each label is a binary task (1 for present and 0 for absent), which has been manually annotated. This task aims to predict annotations for a given review text.

Predicted Class	**Description**
Adverse Drug Reactions (ADR)	-
Withdrawal Symptom (WD)	-
Effective (EF)	-
Ineffective (INF)	-
Sign/Symptom/Illness (SSI)	Text contains explicit SSI as a result of the drug
Drug Indication (DI)	Text contains SSI that is currently being addressed by the drug

Table 1. The six labels predicted by multi-label classifiers

Selected Features Two features were used for multi-label classification. The first is the preprocessed sentences without any additional features. From here on, references to this task-feature pair will be referred as Task 1 Feature 1. The second contains the drug name added to the start of the review text. This will be referred to as Task 1 Feature 2.

Logistic Regression The logistic regression model performed Term Frequency Inverse Document Frequency (TF-IDF) vectorization on the input text. A multi-output classifier was built upon a logistic regression model, which was then tuned using hyperparameter tuning.

Transformer The transformer model uses a pre-trained BERT model, which was trained on a downstream task in order to create a multi-label transformer classifier. Binary cross-entropy loss (BCE) with logits (a sigmoid layer) loss was used for loss calculation. BCE is well-suited for binary classification[6] such as this task, and using the logits replaces negative values with 0s to avoid large negative numbers. Table 2 shows the configurations used in the transformer. 15 epochs were done for each transformer model, resulting in convergence of loss and accuracy. The training and validation loss and accuracies are shown in appendix B.

Parameter	**Value**
hidden_size	768
num_hidden_layers	24
num_attention_heads	12
intermediate_size	3072
num_labels	6
optimizer	AdamW
loss_calculation	BCE With Logits Loss

Table 2. BERT Configuration for Multi-Label Classification

Statistical Tests Multi-label classification will be tested using macro averaged precision, recall, and F1-scores. The choice to use macro instead of micro was to give equal weights to each class, as some labels were more prevalent than others (e.g., ADR had a test count of 1712, while INF had a test count of 2548).

Accuracy will also be reported. In addition, following figure 1, the Hamming Loss (HL) of the predictions are also calculated. HL measures the fraction of labels that are incorrectly predicted, on average, across all samples and is used to evaluate multi-label classification tasks[7]. HL ranges from 0 to 1, where 1 means all predictions are erroneous, while 0 means perfect predictions.

Statistical test equations are found in Appendix A.

Multi-Class Classification

Multiclass classification aims to take in text and predict the drug that is being reviewed. There are four classes of drugs to predict. Lexapro, cymbalta, effexorxr, and zoloft.

Selected Features Three features were selected for multi-class classification. The first is the preprocessed sentence without any additional features, referred to as Task 2 Feature 1.

The second is the preprocessed text along with its annotations. For the logistic regression model, the binary annotations are simply added onto the end of the sentences as 1s and 0s.

The transformer's input has the review text concatenated with the predicted class names. If the class is labelled 1, a [POS] token was placed in front of it. If the class is labelled 0, a [NEG] token was placed instead (e.g., if the label EF was labelled 1, this would be transformed into "[POS] effective"). These tokens identify positive(1) and negative(0) labels respectively. This feature will be referred to as Task 2 Feature 2. References to this task-feature pair will be described with the model (e.g., Task 2 Feature 2 Transformer) when the specific model feature is of importance.

The third feature is to only use the annotation inputs. This will be referred to as Task 2 Feature 3. The logistic regression model takes in only the binary annotations, while the transformer only takes in the predicted class name along with the described tokens.

Logistic Regression TF-IDF was also used on the input text. However, in contrast to using a multi-label classifier built on top of a logistic regression model, multi-class classification uses logistic regression directly.

Transformer The transformer model is identical to the multi-label transformer, except it was trained on a multi-class downstream task. The number of hidden layers was also decreased due to long training times. The loss calculation was also changed to cross-entropy as it is a better fit for classification tasks[8]. The transformer specifications are shown in table 3. 20 epochs were done for each transformer model, resulting in convergence of loss and accuracy. The training and validation loss and accuracies are shown in appendix B.

Parameter	**Value**
hidden_size	768
num_hidden_layers	8
num_attention_heads	12
intermediate_size	3072
num_labels	1
optimizer	AdamW
loss_calculation	Cross-Entropy Loss

Table 3. BERT Configuration for Multi-Class Classification

Statistical Tests Multi-class classification will be tested using macro averaged precision, recall, and F1-scores as described previously. The use of macro averaging was due to class imbalance (Zoloft had a test count of 565 while cymbalta had a test count of 791). Accuracy will also be reported.

Additionally, following figure 1, Cohen's Kappa (Cohen's K) and Matthews Correlation Coefficient (MCC) will be used. Cohen's K is used to calculate inter-model agreement on predictions. It returns a value between 1 and -1, where 1 means a perfect agreement, 0 means no agreement above chance, and -1 indicating less agreement than random chance. Cutoff points for Cohen's K is given in table 4, which is based on McHugh's research[9].

Absolute Cohen's Kappa Range	**Interpretation**
$\lvert\kappa\rvert \leq 0$	No agreement
$0.01 \leq \lvert\kappa\rvert \leq 0.20$	None to slight agreement
$0.21 \leq \lvert\kappa\rvert \leq 0.40$	Fair agreement
$0.41 \leq \lvert\kappa\rvert \leq 0.60$	Moderate agreement
$0.61 \leq \lvert\kappa\rvert \leq 0.80$	Substantial agreement
$0.81 \leq \lvert\kappa\rvert \leq 1.00$	Almost perfect agreement

Table 4. Cohen's Kappa Cutoff Points

MCC is generally used to measure the quality of binary classifications by comparing predictions against their ground truth[10]. However, it has been adapted for multi-class classification by taking into account all the true and false positive as well as true and false negatives for each class[11]. This adaptation of the MCC can indicate whether the model is effectively predicting each class, considering the imbalance between classes and overall performance across all classes.

MCC returns a value betwen 1 and -1, where 1 means perfect predictions, 0 means a prediction that is no better than random, and -1 means total disagreement between predictions and ground truth. MCC follows cutoff point selection of graphs such as an area under a recciever operating characteristic (AUROC) curve[12]. These cutoff points are generally calculated by generating a graph that plots sensitivity (true positives) against specifictiy (1 - false positives), before testing discriminative cutoff values. Additional details on this process can be found in Yang's research[13]. However, this report will follow an arbitrary cutoff point described in table 5.

Absolute MCC Values	**Interpretation**
$0 \leq \lvert\text{MCC}\rvert \leq 0.1$	Very Poor Predictions
$0.1 < \lvert\text{MCC}\rvert < 0.3$	Poor Predictions
$0.3 \leq \lvert\text{MCC}\rvert < 0.5$	Moderate Predictions
$\lvert\text{MCC}\rvert \geq 0.7$	Good Predictions

Table 5. MCC Cutoff Points

Friedman's Test of Significance

In order to investigate whether the predictions are significantly different from each other, Friedman's test will be attempted to be performed on both tasks. Friedman's test is a hypothesis testing method, where the null hypothesis is that there is no significant difference between two samples of predictions[14]. Meanwhile, the alternative hypothesis is that a significant difference does exist. This test returns a probability(p)-value, which indicates how likely is it to get result predictions if the compared samples have no significant differences. A cutoff point (α value) of 0.05 will be used, meaning that if the p-value is <0.05, the null hypothesis will be rejected. A p-value of <0.05 means that there is less than a 5% chance that the obtained results were due to random chance.

This test was found to not be suitable for the multi-class classification task, due to the output. Additional discussion on the impact of this will be discussed in the discussion and conclusion section.

For the multi-label classification, each label had the Friedman's test performed to investigate whether the predictions are significantly different from each other for each label.

- **Null Hypothesis (H_0):** There is no significant difference between the models' predictions for the label.
- **Alternative Hypothesis (H_1):** There is a significant difference between the models' predictions for the label.
- $\alpha = 0.05$

Results and Analysis

Results for multi-label (task 1) and multi-class (task 2) classifications are shown and discussed below.

Multi-Label Statistical Tests

Accuracy, Precision, Recall, and F1-Scores Table 6 shows the accuracy along with the macro precision, recall, and F1 scores. Transformer models also show training and validation loss.

Model	**Accuracy**	**Macro Precision**	**Macro Recall**	**Macro F1**	**Training Loss**	**Validation Loss**
Task 1 Feature 1 Transformer	0.98	0.95	0.91	0.93	0.0134	0.0789
Task 1 Feature 1 Logistic Regression	0.5083	0.64	0.38	0.47	N/A	N/A
Task 1 Feature 2 Transformer	0.98	0.92	0.93	0.92	0.0115	0.0772
Task 1 Feature 2 Logistic Regression	0.5249	0.65	0.38	0.47	N/A	N/A

Table 6. Accuracy and Macro-Averaged Precision, Recall, F1-Scores, Training Loss, and Validation Loss for Task 1

Transformers show much higher performances compared to logistic regression for both features. Change in features does not appear to significantly affect model performance for all models, with the highest change being accuracy in logistic regression models (1.66% difference).

Hamming Loss The results of calculating HL for multi-label classification are shown in table 7. Each model was compared to the ground truth.

Model	**Hamming Loss Against Ground Truth**
Task 1 Feature 1 Transformer	0.016057586
Task 1 Feature 1 Logistic Regression	0.11517165
Task 1 Feature 2 Transformer	0.019379845
Task 1 Feature 2 Logistic Regression	0.109634551

Table 7. Hamming Loss for Multi-Label Task (Task 1)

The transformer models have significantly lower HL than the logistic regression models. This is as expected, as a higher accuracy indicates a higher chance of correct predictions. However, HL can show a more detailed view on model performance. For example, feature 1 using logistic regression has an accuracy of 50.83%. This accuracy metric is strict as it only considers an instance as correct if all labels were correctly predicted. However, the HL for this model suggests that only about 11.16% of the predictions were incorrect, indicating that the model performs better when considering individual label predictions.

Friedman's Test of Significance Friedman's test was done on all the models simultaneously. It returns two values, a test statistic and a p-value. Results are shown in table 8.

Label	Statistic	p-value
ADR	16.373	0.003
WD	10.667	0.031
EF	64.681	3.00×10^{-13}
INF	12.653	0.013
SSI	34.712	5.32×10^{-7}
DI	27.855	1.33×10^{-5}

Table 8. Friedman's Test of Significance for Multi-Label Task (Task 1)

The results of the Friedman's test statistic shows the difference in predictions for all models. The EF label showed the highest amount of difference, while WD showed the least. These differences are statistically significant, as each of the p-values are less than the determined cutoff point (0.05). As such, we reject the null hypothesis. These results provide evidence in favor of the alternative hypothesis that each model's predictions are significantly different.

Multi-Class Statistical Tests

Accuracy, Precision, Recall, and F1-Scores Table 9 shows the accuracy along with the macro precision, recall, and F1 scores. Transformer models also show training and validation loss.

Model	Accuracy	Macro Precision	Macro Recall	Macro F1	Training Loss	Validation Loss
Task 2 Feature 1 Transformer	0.84	0.84	0.84	0.84	0.044	0.286
Task 2 Feature 1 Logistic Regression	0.3688	0.78	0.35	0.29	N/A	N/A
Task 2 Feature 2 Transformer	0.834	0.84	0.84	0.83	0.031	0.289
Task 2 Feature 2 Logistic Regression	0.3688	0.78	0.35	0.29	N/A	N/A
Task 2 Feature 3 Transformer	0.322	0.33	0.31	0.26	0.696	0.696
Task 2 Feature 3 Logistic Regression	0.3422	0.28	0.31	0.24	N/A	N/A

Table 9. Accuracy and Macro-Averaged Precision, Recall, F1-Scores, Training Loss, and Validation Loss for Task 2

The resulting predictions of the model for multi-class classification generally show a high accuracy for transformers, compared to their logistic regression baselines. However, feature 3 showed a different trend, where the logistic regression baseline performed better. There was also a large change in accuracy compared to other transformer models (with feature 3 having roughly 50% lower accuracy than other transformer models). This could be due to how transformers require rich language information, which feature 3 does not provide, as the inputs are only a series of token-word pairs. Meanwhile, logistic regression might perform better for feature 3 as it does not depend on contextual language understanding, and instead focus on the patterns provided by the input.

Cohen's Kappa A heatmap of Cohen's K for each model tested against each other is shown in figure 2.

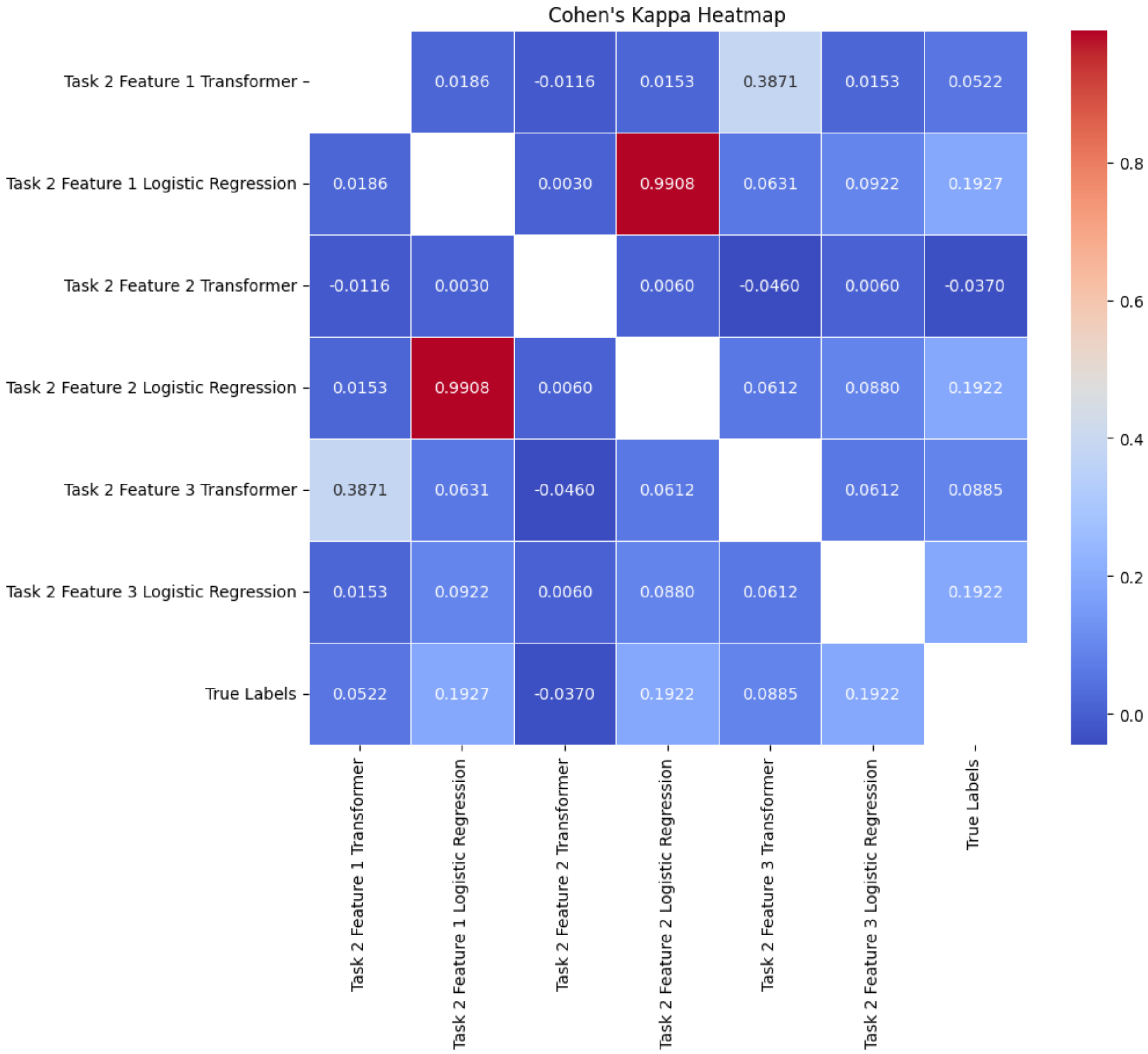

Figure 2. Heatmap of Cohen's Kappa Results on Multi-Class Classification Task

The results of Cohen's K shows that for most models, the agreement between predictions are low. This is true even for the models with high accuracy. For example, the transformer models for features 1 and 2 have similar accuracies (84% and 83.4% respectively). However, they have a Cohen's K value of -0.0116, which, following our defined cutoff points, is no more than random chance. This indicates that on the wrong predictions, these models do not predict the same incorrect class even if they have high accuracy on correct predictions.

Between all these models, there are only 2 Cohen's K values that are greater than fair agreement. The first is between the logistic regression models of feature 1 and 2, which has a Cohen's K of 0.99, indicating an almost perfect positive agreement. This could be due to how the inputs for feature 1 and 2 are similar, and logistic regression models predict similar patterns for each of these. As such, the models predict similar classes, even for incorrect predictions.

The second is between the transformer models of feature 1 and feature 3 with a Cohen's K of 0.39, indicating substantial positive agreement. Upon inspection, predictions of the transformer models for feature 3 only predict mainly one class. This could cause a higher than expected agreement with the transformer model for feature 1, depending on how the model for feature 1 predicts incorrect predictions.

Matthews Correlation Coefficient Each model was compared against the ground truth to calculate their MCC values. The resulting values are shown in table 10.

	Task 2 Feature 1 Transformer	**Task 2 Feature 1 Logistic Regression**	**Task 2 Feature 2 Transformer**	**Task 2 Feature 2 Logistic Regression**	**Task 2 Feature 3 Transformer**	**Task 2 Feature 3 Logistic Regression**
MCC Score	0.066509038	0.194445905	-0.043092423	0.19418937	0.103772985	0.124725893

Table 10. MCC Scores for Each Model, Compared Against Ground Truth

The results of the MCC indicate that most models are not able to correctly predict across all classes evenly, even for models that have high accuracy. For example, the transformer models for features 1 and 2 have an accuracy of 84%. However, they have MCC scores that indicate very poor predictions when considering performance across all classes. This could suggest that the model performs significantly worse when considering a class-to-class basis.

However, for some models that do not have high accuracy, they appear to have higher MCC scores. For example, the logistic regression models have an MCC score of about 0.12-0.19, and even the model that predicts only one class(transformer for feature 3), has an MCC score of 0.10. This particular transformer model predicted only the majority class, allowing it to minimize false positives and false negatives. As such, although they do not have a good accuracy, they are able to minimize misclassifications, leading to a decent MCC score.

Discussion

This report has performed the statistical tests suggested by Rainio, Tauho, and Klén[5] on two natural language processing tasks, namely a multi-label binary annotation task and a multi-class drug classification task. For each task, different statistical tests were applied to investigate model behaviour.

Multi-Label Classification Task

Interpretation of the statistical tests on the multi-label task can be considered in three stages. First, the traditional machine learning metrics show that transformer models are able to outperform their logistic regression baselines consistently. Furthermore, the difference in features affect logistic regression and transformer models differently, albeit only slightly. Secondly, the HL values provide further support that transformers are able to outperform logistic regression models, even when considering individual label predictions. However, the HL shows that logistic regression models perform better on a label-by-label basis than suggested by their accuracy, as HL relaxes the requirement that all the labels are correctly predicted for an instance to be considered correct. Finally, Friedman's test shows that the difference in model predictions are statistically significant.

Multi-Class Classification Task

Interpretation of the multi-class classification task shows opposite performance when comparing traditional machine learning metrics to statistical tests. Traditional machine learning metrics show that for features 1 and 2, the transformer outperforms their logistic regression baselines. However, for feature 3, the transformer performed worse than their baseline. This behaviour highlights the difference in expected behaviours. Logistic regression learns patterns, while transformers capture language understanding, and depending on the input, learning simpler patterns could lead to better performance.

Investigation on Cohen's K also further supports how logistic regression models learn similar patterns given similar inputs, while transformers might learn different weights with similar inputs. This leads to different incorrect predictions. Furthermore, Cohen's K can also provide insights into what incorrect predictions are done. For example, the transformer models of feature 1 and 3 have a substantial positive agreement, indicating that feature 1 might default to the majority class for incorrect predictions. While this report does not investigate this behaviour in depth, the results of Cohen's K provides an indicator of possible model behaviour that could be investigated further.

Finally, the results of the MCC scores show that transformer models do not tend to each class equally, which could support the behaviour of the transformer for feature 1 to default to a majority class. Similar low MCC scores could also indicate similar behaviour. While investigating the exact causes of these MCC scores are out of the scope of this report, MCC scores can provide indications on what the model struggles with, such as evenly predicting every class.

Friedman's Test on Multi-Class Classification

The application of Friedman's test on multi-class classification poses a challenge due to the output, which comes in the form of text. Friedman's test calculates differences between a series of predictions. In multi-label classification, this difference is calculated as either 1 or 0 (with 1 meaning that there is a difference between predictions and 0 as having the same predictions). However, text data such as drug names need to be preprocessed into numerical variables. While some studies have used Friedman's test on textual data by converting it into a ranking task[15], there is no consensus on how to handle this change in features. Some methods that could be investigated could involve converting text into a word embedding such as TF-IDF, calculating the vector differences to identify distances between predictions. However, this is out of the scope of this report, but future work could be done to tackle this issue.

Conclusion and Future Directions

In conclusion, this report has shown how statistical tests, combined with traditional machine learning metrics can help to better understand model behaviour. However, it should be noted that each metric performs different functions. While statistical tests indicate model behaviour, traditional machine learning models are still important when considering practicality. For example, for multi-class classification, while the transformer models for feature 1 and 2 do not perform evenly on all classes, their accuracy would still make a transformer model preferred in performing predictions.

In addition, while statistical tests suggest possible undesireable model behaviour, these metrics could still be misleading, such as how predicting a majority class inflates the MCC score. While some studies have suggested how MCC has advantages over F1-scores and accuracy[12], these metrics should still be understood as providing ideas on model behaviour, instead of replacing traditional machine learning metrics.

Future work can investigate more sophisticated workflows to understand model behaviour based on statistical tests. In addition, the difficulty in using some statistical tests on textual data such as Friedman's test described above could be further investigated. Additional statistical tests could also be integrated, such as testing assumptions of independence, which could guide what statistical tests fit a specific task and to identify possible problems with a provided dataset[16].

References

1. Zolnoori M, Fung K, Patrick T, Fontelo P, Kharrazi H, Faiola A, et al. The PsyTAR Dataset: From Patients Generated Narratives to a Corpus of Adverse Drug Events and Effectiveness of Psychiatric Medications. Data in Brief. 2019 03;24:103838.

2. Zolnoori M, Fung KW, Patrick TB, Fontelo P, Kharrazi H, Faiola A, et al. A systematic approach for developing a corpus of patient reported adverse drug events: A case study for SSRI and SNRI medications. Journal of Biomedical Informatics. 2019;90:103091. Available from: https://www.sciencedirect.com/science/article/pii/S1532046419300012.

3. Dror R, Baumer G, Shlomov S, Reichart R. The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing. In: Gurevych I, Miyao Y, editors. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Melbourne, Australia: Association for Computational Linguistics; 2018. p. 1383-92. Available from: https://aclanthology.org/P18-1128.

4. Wang X, Hripcsak G, Markatou M, Friedman C. Active Computerized Pharmacovigilance Using Natural Language Processing, Statistics, and Electronic Health Records: A Feasibility Study. Journal of the American Medical Informatics Association. 2009 05;16(3):328-37. Available from: https://doi.org/10.1197/jamia.M3028.

5. Rainio O, Tauho J, Klén R. Evaluation metrics and statistical tests for machine learning. Sci Rep. 2024;14(6086).

6. Xu Z, Liu R, Yang S, Chai Z, Yuan C. Learning Imbalanced Data with Vision Transformers; 2023.

7. Díez J, Luaces O, del Coz JJ, Bahamonde A. Optimizing different loss functions in multilabel classifications. Regular Paper. 2015;3:107-18. Available from: https://link.springer.com/article/10.1007/s13748-014-0060-7.

8. Hui L, Belkin M. Evaluation of Neural Architectures Trained with Square Loss vs Cross-Entropy in Classification Tasks; 2021.

9. McHugh ML. Interrater reliability: the kappa statistic. Biochem Med (Zagreb). 2012;22. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3900052/#:~:text=Cohen%20suggested%20the%20Kappa%20result,1.00%20as%20almost%20perfect%20agreement.

10. Chicco D, Tötsch N, Jurman G. The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. BioData Mining. 2021;13. Available from: https://doi.org/10.1371/journal.pone.0177678.

11. Jurman G, Riccadonna S, Furlanello C. A Comparison of MCC and CEN Error Measures in Multi-Class Prediction. PLOS ONE. 2012 08;7(8):1-8. Available from: https://doi.org/10.1371/journal.pone.0041882.

12. Chicco D, Jurman G. The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification. BioData Mining. 2023 02;16.

13. Yang S, Berdine G. The receiver operating characteristic (ROC) curve. The Southwest Respiratory and Critical Care Chronicles. 2017 May;5(19):34-6. Available from: https://pulmonarychronicles.com/index.php/pulmonarychronicles/article/view/391.

14. Hoffman JIE. Chapter 26 - Analysis of Variance II. More Complex Forms. In: Hoffman JIE, editor. Biostatistics for Medical and Biomedical Practitioners. Academic Press; 2015. p. 421-47. Available from: https://www.sciencedirect.com/science/article/pii/B9780128023877000263.

15. Wang J, Kan H, Meng F, Mu Q, Shi G, Xiao X. Fake Review Detection Based on Multiple Feature Fusion and Rolling Collaborative Training. IEEE Access. 2020 01;8:182625-39.

16. Flatt C, Jacobs R. Principle Assumptions of Regression Analysis: Testing, Techniques, and Statistical Reporting of Imperfect Data Sets. Advances in Developing Human Resources. 2019 11;21:484-502.

Appendix

Appendix A - Equations for Statistical Tests

Cohen's K:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{1}$$

where:

$$\begin{aligned} p_o &= \text{relative observed agreement among raters} \\ p_e &= \text{hypothetical probability of chance agreement} \end{aligned}$$

(p_o) is calculated as:

$$p_o = \frac{a + d}{a + b + c + d} \tag{2}$$

(p_e) is calculated as:

$$\begin{aligned} p_e = &\left(\frac{(a+b)}{a+b+c+d} \cdot \frac{(a+c)}{a+b+c+d} \right) \\ &+ \left(\frac{(c+d)}{a+b+c+d} \cdot \frac{(b+d)}{a+b+c+d} \right) \end{aligned} \tag{3}$$

where:

- a = Number of times both annotators agreed the sample was positive
- d = Number of times both annotators agreed the sample was negative
- b = Number of times Annotator A said positive but Annotator B said negative
- c = Number of times Annotator A said negative but Annotator B said positive

Matthews Correlation Coefficient:

$$\text{MCC} = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4}$$

$$\begin{aligned} &\text{where:} \\ &TP : \text{True Positive} \\ &FP : \text{False Positive} \\ &TN : \text{True Negative} \\ &FN : \text{False Negative} \end{aligned}$$

Hamming Loss:

$$\text{HL} = \frac{1}{N \times L} \sum_{l=1}^{L} \sum_{i=1}^{N} (Y_{i,l} \neq X_{i,l})$$

$$\begin{aligned} &\text{where:} \\ N &: \text{Number of samples} \\ L &: \text{Number of labels} \\ Y_{i,l} &: \text{True label for } i\text{th sample and} \\ &\quad l\text{th label} \\ haty_{ij} &: \text{Predicted label for } i\text{th sample and} \\ &\quad l\text{th label} \\ (Y_{i,l} \neq haty_{ij}) &: 1 \text{ if } y_{ij} \neq \hat{y}_{ij} \text{ and 0 otherwise} \end{aligned}$$

Accuracy, Precision, Recall, and F1-Scores:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{8}$$

$$\begin{aligned} &\text{where:} \\ TP &: \text{True Positive} \\ FP &: \text{False Positive} \\ TN &: \text{True Negative} \\ FN &: \text{False Negative} \end{aligned}$$

Friedman's Test:

$$T = \frac{12 \sum S_{ij}^2}{jk(j+1)} - 3k(j+1) \tag{9}$$

where:

$$\begin{aligned} T &= \text{test statistic} \\ \sum S_{ij}^2 &= \text{sum of the squared sums of ranks for each prediction set} \\ j &= \text{number of prediction sets} \\ k &= \text{number of instances} \end{aligned}$$

Appendix B - Transformer Model Performances

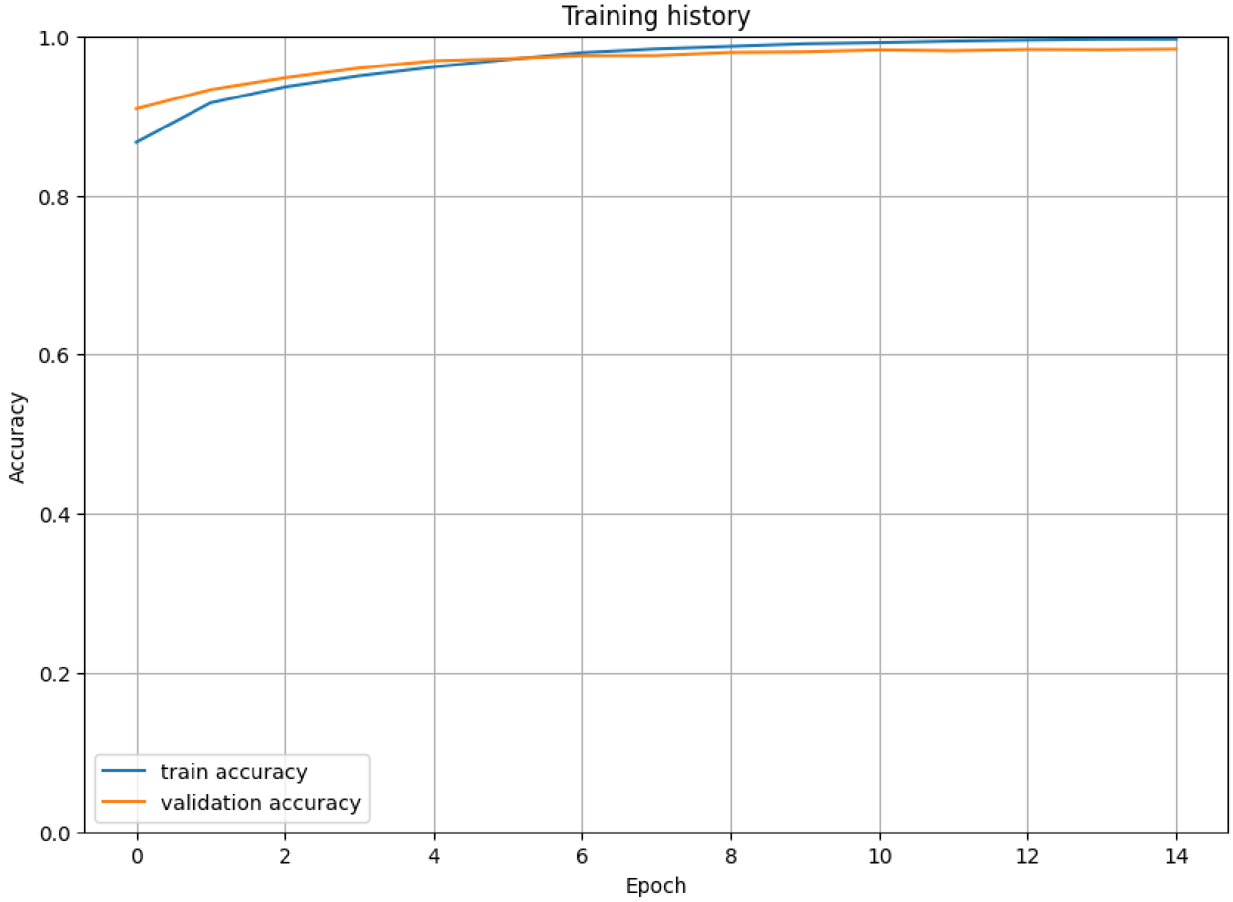

Figure 3. Task 1 Feature 1 Model Training (Total Training Time: 2056.51 Seconds)

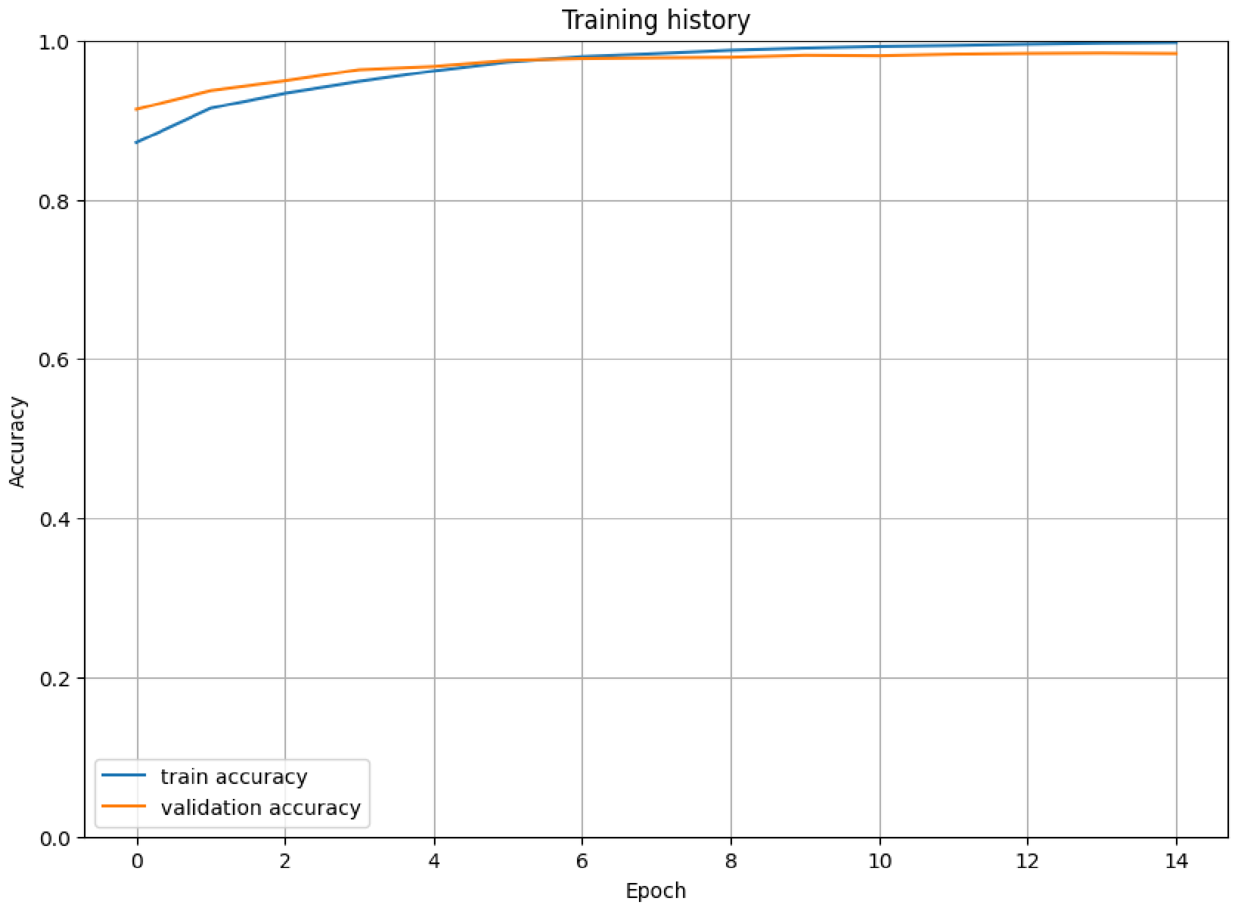

Figure 4. Task 1 Feature 2 Model Training (Total Training Time: 2037.76 Seconds)

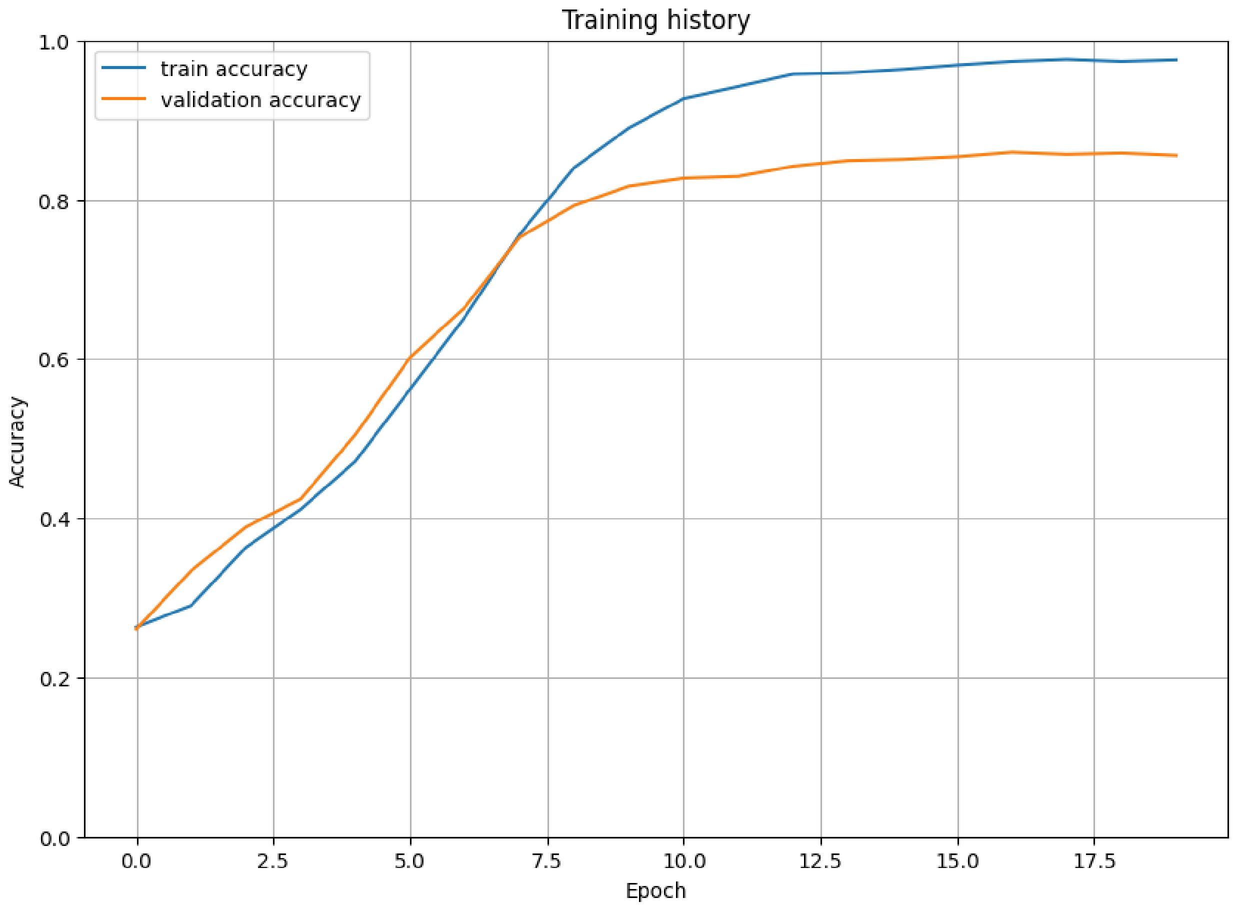

Figure 5. Task 2 Feature 1 Model Training (Total Training Time: 7471.89 Seconds)

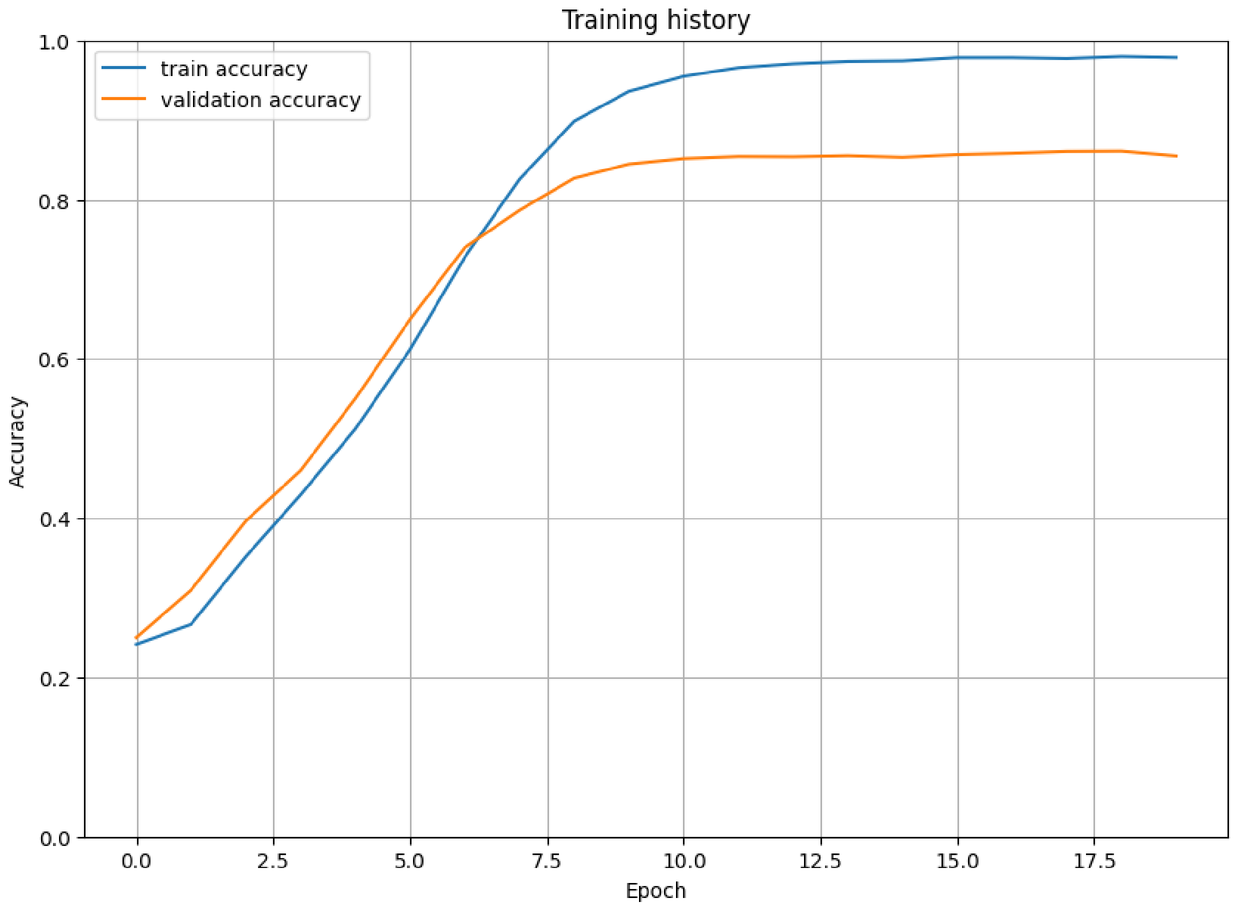

Figure 6. Task 2 Feature 2 Model Training (Total Training Time: 6587.32 Seconds)

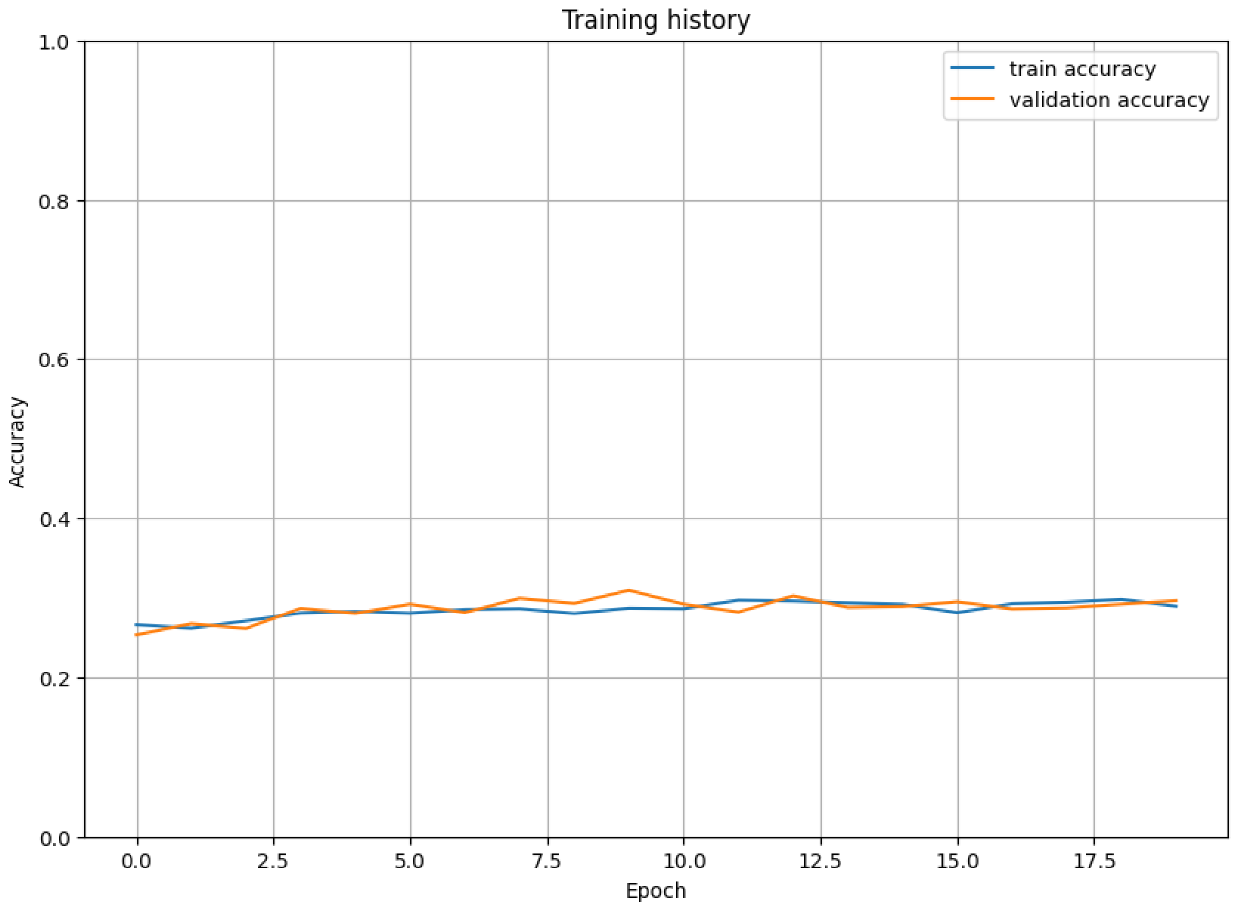

Figure 7. Task 2 Feature 3 Model Training (Total Training Time: 6590.17 Seconds)

Appendix C - Source Code