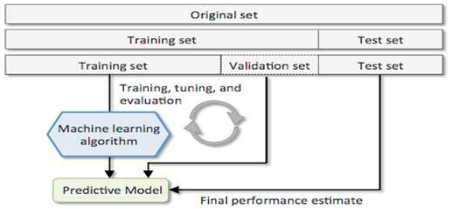
Generating Textual Explanations for Machine Learning Models Performance: A Table-to-Text Task

Isaac K. E. Ampomah[1], James Burton[1], Amir Enshaei[2], Noura Al Moubayed[1]
[1]Department of Computer Science, Durham University, [2]Caspian Learning

Background and Objectives



A typical classification model development flow
Source: https://vitalflux.com/hold-out-method-for-training-machine-learning-model/

- A typical machine learning (ML) model development flow is shown in the figure above.
- Numerical tables are widely employed to communicate or report the classification performance of machine learning (ML) models with respect to a set of evaluation metrics. For non-experts, domain knowledge is required to fully understand and interpret the information presented by numerical tables.
- This paper proposes a new natural language generation (NLG) task where neural models are trained to generate textual explanations, analytically describing the classification performance of ML models based on the metrics' scores reported in the tables.
- Presenting the generated texts along with the numerical tables (as shown below) will allow for a better understanding of the classification performance of ML models.

Classification Performance Summary

Dataset	Imbalanced (62% and 38%)			
Labels	C1 and C2			
	Metrics			
	Sensitivity	Precision	Accuracy	AUC
Value	90.32	89.13	90.73	95.87

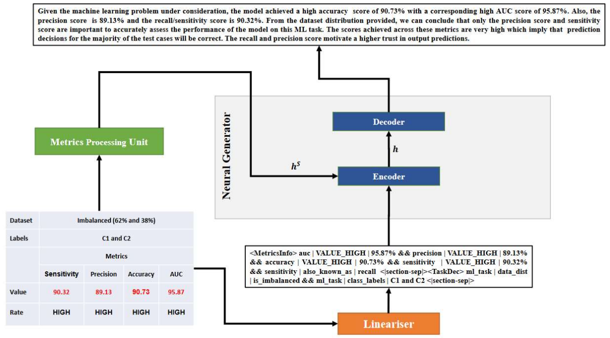
Textual Summary

Given the machine learning problem under consideration, the model achieved a high accuracy score of 90.73% with a corresponding high AUC score of 95.87%. Also, the precision score is 89.13% and the recall/sensitivity score is 90.32%. From the dataset distribution provided, we can conclude that only the precision score and sensitivity score are important to accurately assess the performance of the model on this ML task. The scores achieved across these metrics are very high which imply that prediction decisions for the majority of the test cases will be correct. The recall and precision score motivate a higher trust in output predictions.

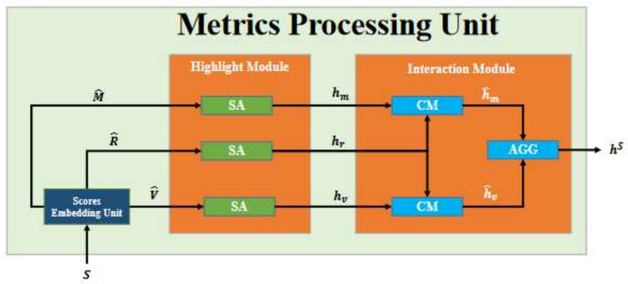
An example of a classification performance report table (containing the score with respect to each metric along with information on the distribution of the underlying data across the two classes: C1 and C2) and the corresponding textual explanation.

Model Performance Narration Model

- Our experiment and analysis are based on the T5 (Raffel et al., 2020) and BART (Lewis et al., 2020) language models.
- The input to the NLG models is a numerical table containing the evaluation metrics' scores of a classifier, the list of class labels and the distribution of the dataset across the labels.
- The performance report tables are converted to flat-strings by a linearizer based on a pre-defined template.
- To improve the quality of the narrative generated, the neural generator as shown below is augmented with semantic representations of the metrics information generated by a neural module: **Metrics Processing Unit (MPU)**.



An illustration of our table-to-text neural generator trained to produced the textual explanation based on the linearised table and metrics-ratings-values semantic representation, h^s, learned by the Metrics Processing Unit (MPU)



Architecture of the Metrics Processing Unit (MPU) employed to learn to h^s from the list of performance scores S.

Experimental Setup: Dataset, Model Settings, Training, and Inference

- The dataset for this task comprises 825 analytical textual explanations written by experts summarizing the performance of ML models trained on 59 classification tasks across different application domains. 100 data-narrative pairs are sampled as the test set.
- Permutating the order in which the evaluation metrics and their corresponding scores are passed to the models increased the training set size from 725 to 4529.
- The fine-tuning of the T5 and BART models is performed using the Adam optimizer [Kingma and Ba, 2014] with an initial learning rate of 3e-4.
- For simplicity, the T5-variant+MPU and BART-variant+MPU respectively, denote the T5 and BART model variants augmented with the auxiliary information from the MPU.

Experimental Results

Model	BLEU	METEOR	PARENT	BLEURT
T5-small	37.83±2.25	39.46±1.83	32.12±1.16	55.52±0.22
T5-small + MPU	40.58±1.95	43.40±1.05	**34.04±0.65**	**56.15±0.40**
T5-base	46.31±0.89	47.45±0.35	30.99±0.7	54.09±0.30
T5-base + MPU	45.46±0.60	47.75±0.40	31.33±0.67	54.46±0.19
T5-large	45.70±0.73	47.11±0.31	31.08±0.63	54.53±0.2
T5-large + MPU	46.03±0.75	47.53±0.41	31.58±0.73	54.63±0.42
BART-base	44.83±1.03	**47.83±0.22**	32.79±0.86	54.71±0.16
BART-base + MPU	45.75±0.84	**47.83±0.64**	33.26±0.96	55.26±0.31
BART-large	45.57±1.96	46.41±1.44	32.29±2.12	51.29±1.76
BART-large + MPU	**46.98±0.73**	47.52±1.50	33.27±0.6	51.25±1.67

Evaluation of generation performance of the neural models on the dataset. "+ MPU" detonates training the variant of the pre-trained base-lines with Metrics Processing Unit (MPU). The models are fine-tuned with five different random seed and the scores are based on the average and standard deviation.

- The performance of the models are evaluated based on the metrics: **BLEU, METEOR, BLEURT, and PARENT**.
- The different variants of the pre-trained models achieved varying scores.
- Augmenting the linearised representation with the semantic representations of the metrics information in the table with MPU is shown to further improves the generation performance of the underlying models. Example compare the performance of T5-small and T5-small+MPU models.

Examples of Generated Model Performance Narrations

Model	Generated
Table	<MetricsInfo>auc	VALUE_HIGH	95.87% && precision	VALUE_HIGH	89.13% && accuracy	VALUE_HIGH	90.73% && sensitivity	VALUE_HIGH	90.32% && sensitivity	also_known_as	recall <|section-sep|><TaskDec>ml_task	data_dist	is_imbalanced && ml_task	class_labels	C1 and C2 <|section-sep|> <|table2text|>
Reference	Given the machine learning problem under consideration, the model achieved a high accuracy score of 90.73% with a corresponding high AUC score of 95.87%. Also, the precision score is 89.13% and the recall/sensitivity score is 90.32%. From the dataset distribution provided, we can conclude that only the precision score and sensitivity score are important to accurately assess the performance of the model on this ML task. The scores achieved across these metrics are very high which imply that prediction decisions for the majority of the test cases will be correct. The recall and precision score motivate a higher trust in output predictions.
T5-small	The performance of the classifier on this binary classification task as evaluated based on the precision, AUC, accuracy, and sensitivity achieved the scores 89.13%, 90.32% , 95.87%, and 90.73% respectively. These scores support the conclusion that this model will be highly effective at correctly classifying most test cases/samples with only a few misclassifications. Furthermore, the confidence in predictions related to the label C2 is very high.
T5-small+MPU	The algorithm's ability to correctly classify test samples as either C1 or C2 was assessed based on the metrics accuracy, AUC, precision, and sensitivity. It scored 90.73% , 95.87%, 89.13% and 90.32%, respectively. As shown in the table, we can confirm that the algorithm has a very high classification performance and will be very effective at correctly labeling most test cases drawn from any of the different classes.
T5-base	On this imbalanced classification task, the trained model scored 90.32% (recall), 90.73% (accuracy), and 95.87% (AUC). From the precision score, it is obvious that the model has a moderately high false-positive rate hence the confidence in predictions related to the C2 label is very high. However, based on the sensitivity and precision scores, we can see that some examples belonging to C1 are likely to be misclassified as C2. This is further supported by the high accuracy and AUC scores.
T5-base+MPU	The classifier scored close to perfect scores across all the metrics (i.e. Precision, AUC, Accuracy and Sensitivity). From the results table, we can see that it scored 90.73% (accuracy), 90.32% (sensitivity), and 95.87% (AUC). Surprisingly, these scores were achieved even though the dataset was imbalanced. With such high scores for precision and sensitivity, the model is shown to have a lower misclassification error rate. Overall, this model has relatively high classification performance, and hence will struggle to correctly identify the labels for only a few test cases belonging to the different classes.

Sample Narrative output from the T5 small and base model variants with and without the MPU

- As shown, the neural generators can produce high-quality classification performance explanations capturing the information presented in the input structured data. There were some cases where the generator fails (highlighted in red) to accurately verbalise the content of the performance metric table.
- However, the metrics-values-ratings contextual information from the MPU allow the pre-trained language models to generate accurate analytical textual explanations based on the related table.

C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21:1–67, 2020.
Lewis, Mike, et al. "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension." Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020.
D. P. Kingma and J. Ba. Adam: A method for stochastic optimization.arXiv:1412.6980, 2014.