Prognosticating Colorectal Cancer Recurrence using an Interpretable Deep Multi-view Network

Danliang Ho HO.DANLIANG@U.NUS.EDU
National University of Singapore, Singapore

Iain Bee Huat Tan IAIN.TAN.B.H@SINGHEALTH.COM.SG
National Cancer Center Singapore, Singapore

Mehul Motani MOTANI@NUS.EDU.SG
National University of Singapore, Singapore

Abstract

Colorectal cancer (CRC) is among the top three most common cancers worldwide, and around 30-50% of patients who have undergone curative-intent surgery will eventually develop recurrence. Early and accurate detection of cancer recurrence is essential to improve the health outcomes of patients. In our study, we propose an explainable multi-view deep neural network capable of extracting and integrating features from heterogeneous healthcare records. Our model takes in inputs from multiple views and comprises: 1) two subnetworks adapted to extract high quality features from time-series and tabular data views, and 2) a network that combines the two outputs and predicts CRC recurrence. Our model achieves an AUROC score of 0.95, and precision, sensitivity and specificity scores of 0.84, 0.82 and 0.96 respectively, outperforming all-known published results based on the commonly-used CEA prognostic marker, as well as that of most commercially available diagnostic assays. We explain our model's decision by highlighting important features within both data views that contribute to the outcome, using SHAP with a novel workaround that alleviates assumptions on feature independence. Through our work, we hope to contribute to the adoption of AI in healthcare by creating accurate and interpretable models, leading to better post-operative management of CRC patients.

Keywords: multi-view modelling, explainability, prognostication, colorectal cancer

1. Introduction

Colorectal cancer (CRC) represents a major health risk in modern society, and a key concern is the possibility of cancer recurrence. Although surgery is potentially curative, the risk of post-operative recurrence is high, with approximately 30-50% of patients who undergo the procedure eventually developing recurrent disease (Young et al., 2014). Nonetheless, an early diagnosis of recurrence is of significant clinical interest, as depending on the extent of disease, treatment could still be potentially curative, or at least prolong survival and improve quality-of-life (Israel and Kuten, 2007; Adam and Vinet, 2004). Carcinoembryonic antigen (CEA), a blood-based tumour marker, has been recommended as a cost-effective means for early recurrence detection (Locker et al., 2006; Castells et al., 1998; Graham et al., 1998). However, several systematic studies have questioned its prognostic value due to limited sensitivity and specificity, also its effect on reducing patient mortality remains to be proven (Sørensen et al., 2016; Shinkins et al., 2017). Furthermore, improving the diagnostic performance of CEA through measures such as increasing follow-up duration and intensity, or supplementing with other monitoring procedures, adds on to the healthcare burden and potentially increases costs of care. As such, there is significant clinical value in developing a more sensitive tool for accurate detection of CRC recurrence.

While machine learning models have been applied to the task (see Section 2.1), to the best of our knowledge, there is little attempt to develop CRC prognostication models that are both highly accurate and explainable. In our study, we propose to use state-of-the-art machine learning techniques to prognosticate and explain potential factors contributing to CRC recurrence. Our contributions are three-fold:

1. We utilise a multi-view deep neural network, Hybrid Transformer for Multi-view Data (`HTMV`), that

uses known prognostic factors, including CEA measurements, to predict and explain CRC recurrence. Our model is capable of automatically extracting and integrating features from heterogeneous healthcare records. It performs significantly better than the CEA marker alone and achieves a 51.7% increase in AUROC over ColoPrint, one of the most commonly used diagnostic assays in the market (see Section 2.1).
2. We generate Shapley additive explanations (SHAP) for post-hoc explanation and visualizations of factors that contribute to our model's prediction. To our knowledge, we are among the first to focus on model explanation for the task of recurrence prediction using multi-view networks.
3. Lastly, we also describe our approach towards alleviating SHAP's feature independence assumption on time-series datasets. We propose a workaround of the feature independence assumption in two steps: 1) we show how to automatically segment time-series into approximately independent regions, and 2) we utilize the partition masking utility within SHAP to impose constraints on the permutation of features that are highly correlated within each region.

2. Related Work

2.1. Prognostication models in CRC research

Work in this area have traditionally been based on statistical approaches to identify reliable prognostic factors, which could then be used to stratify patients into risk groups for more tailored post-operative surveillance. For example, the commercial assays ColoPrint (Kopetz et al., 2015) and OncoDefender-CRC (Lenehan et al., 2012) utilised a small panel of genomic and clinicopathological variables for risk prediction and stratification. We note that the reported AUROC scores on a validation cohort were rather modest (ColoPrint: 0.626, OncoDefender-CRC: 0.55), as would be expected of models that utilise a limited number of prognostic factors.

Machine learning (ML) has emerged as a popular alternative to traditional modelling, with several studies leveraging on standard ML algorithms for the prediction task (Ting et al., 2020; Achilonu et al., 2021). Recent trends towards ever more complex modelling have also encouraged the use of deep learning (DL) techniques, and many studies in this domain used convolutional neural networks to extract morphological features from histopathological images to predict survival and recurrence (Skrede et al., 2020; Geessink et al., 2019; Jiang et al., 2020).

A significant issue associated with the use of DL techniques is the lack of interpretability. Despite its importance, studies that propose models that are both interpretable and accurate are limited and preliminary, for example Ho et al. (2021) described a highly accurate but uninterpretable model for prognosticating recurrence. In another example, Wulczyn et al. (2021) proposed an interpretable DL-based prognostic model where feature importance scores were obtained by fitting regression models on extracted morphological features. We note that in using regression models for explanation, there was an assumption of feature independence and the existence of a linear relationship between features and scores, which is not necessarily true and may have affected the accuracy of the explanation.

2.2. Multi-view modelling

Several studies have demonstrated benefits to considering data from multiple data sources, such as potential improvements in performance and generalisation capabilities (Sun et al., 2019; Zhao et al., 2017). Nonetheless, multi-view modelling is challenging as decisions have to be made on how to best integrate the data, thus work is this domain is rather limited.

The challenges of multi-view modelling has been explored in some recent works, notably for static datasets that do not encode the notion of time. For example, Chaudhary et al. (2018) predicted survival in liver cancer by integrating multi-omics datasets using an autoencoder. Our work differs from theirs in that they do not consider the time dimension in the data integration problem, while we use temporal data as one of the data views. One of the closest works to ours is that by Chowdhury et al. (2019), where they created a multi-view attention-based architecture capable of integrating electronic health records data encompassing different views to learn a unified patient representation. Our work is similar in that we also utilise attention-based mechanisms to learn feature representations, but unlike their work we also explore how to imbue explainability into our model.

2.3. Explanations for multi-view modelling

The explainable AI literature is rife with methods to explain any model architecture in different ways (Selvaraju et al., 2017; Sundararajan et al., 2017; Fukui et al., 2019). However many of them are developed for

specific datasets and have certain architectural constraints. For example, Grad-CAM (Selvaraju et al., 2017) is contingent on the convolutional layers in a CNN. For multi-view modelling we seek an explanation method that has less architectural constraints, as different components of the network may utilise different data structures. Perhaps unsurprisingly, there is limited ongoing work in explanations for complex multi-view networks.

There are two lines of work that are closest to our problem. The first is that of explanations for time-series, as a significant component of our network hinges on time-series modelling. Methods for time-series-based explanations are limited, out of which visualisation of attention weights is commonly used. (Vinayavekhin et al., 2018; Xu et al., 2018). Nonetheless, there is research showing that weights derived from attention do not provide meaningful explanations (Jain and Wallace, 2019). In the second line of work, model-agnostic methods such as SHAP and LIME do not consider the model architecture and represent possible solutions to our problem (Lundberg and Lee, 2017; Ribeiro et al., 2016). However they assume feature independence, therefore employing these methods naively on time-series data can result in inaccurate explanations (Aas et al., 2021). In our work, we propose a method to alleviate the feature independence assumption in SHAP, and outline our approach in utilising SHAP to generate explanations from multi-view networks.

3. Dataset and pre-processing

3.1. Study cohort

Our study was performed using medical data obtained from a cohort of 882 patients diagnosed with Stage 1-3 CRC, with no evidence of metastatic disease. All patients were referred to a local hospital for post-operative follow-up, following surgical resection of the primary tumour. Informed consent was obtained from all patients prior to study enrollment, and institutional ethics approval was obtained for this study.

3.2. Dataset description

The dataset consisted of the following information collected for each patient: a) tabular data on 65 clinical variables that are potentially prognostic for recurrence, such as demographics, tumour characteristics, molecular profiling results and treatment parameters, and b) laboratory measurement data on post-operative CEA levels, collected at multiple timepoints between date of surgery and date of recurrence or most recent follow-up, whichever was earlier. The median length of follow-up was 40 months at a frequency of between 1 to 3 months on average. The median data points per patient was 14. We de-identified all patients by assigning each patient a unique serial number upon study entry, and we removed all personal identifiers prior to data analysis.

3.3. Pre-processing steps

Dataset pre-processing for both tabular and time-series data was performed in the following manner:

Cleaning and de-duplication We removed errors attributed to misspellings, letter case, extra white space and semantically similar categories.

Imputation of missing data For tabular data, we identified whether data is likely to be Missing-Not-At-Random (MNAR) through domain knowledge. We did not attempt to impute MNAR data but rather, we created a new category '*not_available*' to denote MNAR data. All other missing tabular data was imputed with multiple rounds of MICE (van Buuren and Groothuis-Oudshoorn, 2011). For time-series data, we imputed missing data using linear interpolation.

Feature engineering We mined information from unstructured text fields and added them to our tabular dataset via rule-based text extraction.

Time-series resampling This step was conducted only for DL architectures including `HTMV`. We resampled the data monthly to create evenly-spaced intervals, then zero-padded it to the maximum length of the time-series.

Data transformations We normalized all numerical values via logarithmic transformation to remove skewness, followed by min-max scaling. We transformed all categorical data using one-hot encoding.

Our final dataset had 548 features, split into training, validation and test datasets in the ratio 6:2:2. Our dataset was imbalanced, with the Recurrence class taking up only 23% of the dataset. As such, we performed stratified split to ensure that all classes were proportionately represented.

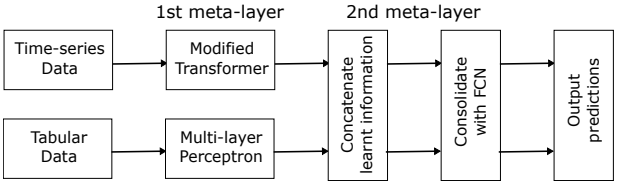


Figure 1: Schematic representation of `HTMV`

4. Proposed model

We design a DL-based architecture that processes and integrates data from time-series and tabular data. Since a significant part of the model is based on the Transformer backbone first proposed by Vaswani et al. (2017), we refer to it as the Hybrid Transformer for Multi-view Data (`HTMV`). A high-level overview of `HTMV` is shown in Figure 1. Our model comprises a two meta-layer architecture:

1. The first meta-layer is composed of two subnetworks, where each network is adapted to extract high-quality features from each data view. We trained a Transformer-based model to model the time-series data, while we utilized an MLP for tabular data modelling.
2. The second meta-layer is an overall network that combines outputs from individual views and learns integrative feature representations to perform the prediction task.

4.1. Time-series modelling using modified Transformer architecture

Originally developed for natural language processing, the Transformer architecture captures long-term dependencies in text data, using dot-product self-attention mechanisms that highlight important pairwise relationships between words (Vaswani et al., 2017). While canonical self-attention has been shown to work well on sequential text, we hypothesized that naively extending the attention mechanism to time-series may impact the Transformer's performance, as the influence of long-range dependencies in time-series data is likely to be less significant as compared to the impact of the direct neighbours surrounding each point. Furthermore, canonical self-attention performs point-wise matching of query-key values and, while useful for learning long-range dependencies, may result in our model becoming less sensitive to the local context.

As such, we explored the following modifications to our Transformer implementation on time-series data. The rest of the model architecture follows that described by Vaswani et al. (2017).

Convolutional self-attention (ConvSA) When learning the attention matrices, instead of creating query, key and value vectors out of Dense layers, we employ 1D-CNNs that convolve across the temporal dimension. This forces the query, key and value vectors to incorporate local context information in the resultant attention calculations, rather than simply timepoint-specific multiplication. We employ causal convolutions to preserve the autoregressive property, with settings of kernel size 3 and stride 1.

Localised attention (LA) We applied a local mask in the decoder that masks out future timesteps and also additionally limits the amount of backward attention, thereby restricting the decoder to only focus on short-term patterns.

We have shown in a previous study that there is an overall beneficial effect to performing both modifications, with a 3 percentage-point increase in AUROC as compared to the unmodified version (Ho et al., 2021).

4.2. Tabular data modelling and feature integration

We extract tabular features using a single hidden layer MLP with 50 hidden nodes, dropout rate set to 0.3 (Srivastava et al., 2014), ReLU activation and He uniform weight initialization (He et al., 2015). We combine features from all subnetworks in the following manner: we obtain fixed-vector representations of features from the last activation layer of each subnetwork (prior to the sigmoid layer), and perform a direct concatenation. We then perform a linear transformation on the integrated signals using a feed-forward network, and subsequently feed them through a sigmoidal function for the final classification.

4.3. Training framework

We create our networks with Tensorflow (Abadi et al., 2015); each was trained on the task of predicting recurrence with the training objective of minimizing binary crossentropy loss. We use RMSProp to optimise weights for the Transformer subnetwork and Adam for the MLP (Kingma and Ba, 2017). Both utilised learning rate decay that started from 1e-4

and decayed at a factor of 0.1 when validation loss failed to improve after 8 epochs. Each model was trained for at least 100 epochs, halting training early when no improvement was observed on the validation loss after 20 epochs.

5. Performance comparisons

5.1. Model evaluation

All developed models were tuned for best hyperparameters on the validation dataset and evaluated on the test dataset. We obtained generalisation performance by first training 20 separate models for each model architecture. Each model was evaluated through 6 bootstrap samples of the test dataset, each sample set to 100 patients with replacement. We then computed sample statistics over all 120 samples and reported the average for the following performance metrics: Precision, Recall (or sensitivity), Balanced Accuracy, Specificity, Area Under Receiver Operating Characteristic (AUROC), and Area under Precision-Recall Curve (AUPRC).

5.2. Baseline models

We designed two categories of baseline models: 1) Shallow ML models that utilise pre-extracted features from time-series data, and 2) Deep temporal networks that employed networks commonly used to analyse time-series data, such as Long Short-Term Memory (LSTM) network and Temporal Convolutional network (TCN). Architectural details of these models are described in Appendix B.

5.3. Results

Table 1 shows the results of comparing model performance between `HTMV` against baseline models. `HTMV` demonstrated the best overall performance, topping the scores in terms of AUROC, AUPRC and Balanced Accuracy. `HTMV` also exhibited a good trade-off between Recall and Specificity at 0.82 and 0.96 respectively, a highly desirable trait when developing models for clinical purposes. This was unlike the other neural network architectures which tended to prioritise one metric over the other. In particular, `LSTM-mlp` did not attain a good balance between Recall and Specificity, with the former topping the charts at 0.85 but the other achieving the lowest score among all models at 0.77. We attribute this to the weighted loss training approach where `LSTM-mlp` was penalised too little for predicting the majority class wrongly. We also note that using the same imbalanced datasets, `HTMV` was able to learn both classes well and that we did not need to additionally cater for an imbalanced learning setup in this model.

6. Explaining `HTMV`

6.1. Overview

We employed Shapley Additive Explanations (SHAP) as an explanation framework to understand the predictions of our black-box neural network. While the Kernel SHAP algorithm is shown to be a computationally efficient approximation to Shapley values on machine learning datasets (Lundberg and Lee, 2017), it assumes that the features are independent, which may result in inaccurate explanations should this assumption not be met. This concern has also been echoed by other studies such as Aas et al. (2021), in which they showed that Kernel SHAP gives less accurate approximations to the true Shapley value on several simulated datasets.

It is evident that our datasets, consisting of both time-series and structured tabular data, does not meet the rigid assumptions of feature independence. In time-series data, adjacent time-points are intrinsically dependent on previous observations, while real-world structured clinical data is known to consist of numerous dependencies due to complex feature interactions. Despite the possibility of obtaining inaccurate explanations, we note that many studies on real-world datasets still proceeded with direct application of SHAP on their use cases (see Saluja et al. (2021) for time-series data and Seki et al. (2021) for tabular data).

It is with such concerns that we are motivated to propose the following workaround that alleviates the feature independence assumption when using SHAP:

- *Time-series segmentation:* We perform automatic segmentation of each time-series into loosely independent regions, using a genetic algorithm approach.
- *Partition masking with SHAP:* Within each region, we enforce structure based on the correlation of the model inputs, by employing SHAP with partition masking. This forces strongly correlated features to be permuted together, preventing the breaking of feature dependencies and therefore generation of unrealistic model inputs.

Table 1: Model comparisons between `HTMV` (our model) with 2 categories of baselines. Results are reported as average of 120 runs (standard deviation).

Models	Precision	Recall	Specificity	Balanced Accuracy	AUROC	AUPRC
Baselines: Shallow ML						
`LR`	0.62 (0.084)	0.71 (0.111)	0.89 (0.038)	0.80 (0.057)	0.90 (0.049)	0.81 (0.084)
`SVM`	0.73 (0.058)	0.73 (0.147)	0.93 (0.015)	0.83 (0.073)	0.91 (0.055)	0.82 (0.093)
`GB`	0.72 (0.081)	0.70 (0.119)	0.93 (0.026)	0.82 (0.062)	0.89 (0.069)	0.79 (0.095)
`RF`	0.53 (0.106)	0.83 (0.106)	0.81 (0.032)	0.82 (0.054)	0.89 (0.064)	0.77 (0.080)
`MLP`	0.77 (0.109)	0.69 (0.121)	0.94 (0.109)	0.82 (0.063)	0.87 (0.067)	0.79 (0.095)
Baselines: Deep temporal networks						
`LSTM-mlp`	0.50 (0.053)	**0.85 (0.099)**	0.77 (0.056)	0.81 (0.050)	0.88 (0.058)	0.72 (0.124)
`TCN-mlp`	**0.86 (0.088)**	0.80 (0.096)	**0.97 (0.023)**	0.88 (0.056)	0.92 (0.053)	0.83 (0.107)
`HTMV`	0.84 (0.091)	0.82 (0.093)	0.96 (0.025)	**0.89 (0.057)**	**0.95 (0.046)**	**0.88 (0.086)**

6.2. Time-series segmentation

We referenced Nikolaou et al. (2015)'s approach to create a genetic algorithm (GA), described in Algorithm 1, that learns the set of optimal cutpoints for time-series segmentation.

Algorithm 1: Time-series segmentation

Input: A single time-series X
Output: chromosome (chr) x_i with highest fitness fit_{x_i}
Initialisation;

1. Randomly segment X to create population pool p_{curr} of N chr with m segments. Each chr x is represented as an array of cutpoints $\{t_1 \ldots, t_{m-1}\}$;
2. Set iteration counter $count = 0$;

while *count < max number of iterations* **do**
  for *i=1,2,..., N/4* **do**
    1. Select two chr with highest fit_{x_i} from p_{curr}.
    2. Apply crossover on pair to generate child
    3. Apply mutation on child at rate p
    4. Add parents and child to new pop pool p_{new}.
  end
  1. Select remaining chr from p_{curr} based on highest fit_{x_i}, adding to p_{new} to make up N.
  2. Compute $fit_{x_i} = S(x_i)$ for x_i in p_{new}.
  3. Let $p_{curr} = p_{new}$.
  4. Increment *count* by 1.
end

The fitness function $S(\cdot)$ is defined as follows. Let m be the number of segments, t_{s-1} and t_s be the index of the first and last data point in m, y_i be the time-series values within each segment, and $\bar{y_s}$ be the average value of the segment. We first calculate the following 6 features for each segment: **variance**, **skewness**, **kurtosis**, **slope of linear regression**, **mean-squared error**, and **autocorrelation coefficient**. We include formulae details within Appendix A.

We then performed k-means clustering on all segments within each chromosome on the 6 features. Our objective was to obtain a clustering that minimizes the sum of squared error (SSE) between the individual segments and their corresponding cluster centroid. Intuitively, a lower SSE is due to the formation of more compact clusters, i.e. segments that are similar on the basis of the 6 features cluster together, while segments that are different cluster far away. Hence, the GA is encouraged to learn the number and location of cutpoints that produces similar segments within each group, while keeping segments across different groups as dissimilar as possible.

Based on experimentation, we set the number of clusters $K = m/2$. To avoid generating clusters each with only a single segment (as SSE would be 0), we thus define our fitness as: $S = m/\text{SSE}$.

6.3. Partition masking with SHAP

We employ the partition masking functionality within SHAP such that strongly correlated features are masked together. Therefore, we represent all features

in a partition tree structure, where leaf nodes are individual features and nodes that are most closely related have the fewest links in between. We create one partition tree for each dataset. For time-series data, we represent individual time-points within each segment at the lowest level of treeA, and the segments at the next level. For tabular data, we consider one-hot encoded categories as analogous to the independent segments within the time-series, and thus represent individual features (e.g. RaceA, RaceB) at the lowest level of treeB, and parent categories (e.g. Race) at the next level.

For our analysis, we permute features together based on correlation within each segment. We also note that this approach can be extended to also consider correlations across segments, represented by scores such as mutual information. Finally, since SHAP with partition masking was designed to only handle a single dataset with a single partition tree, we combine the two trees at the top level and feed this merged version into SHAP to output explanations.

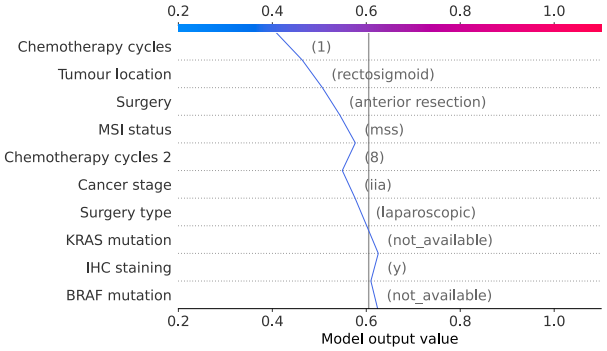
7. SHAP visualisations

In this section, we use the obtained SHAP values to provide individual explanations for both tabular and time-series datasets. We also analyse SHAP values in aggregate to elucidate associations between our features and the recurrence outcome.

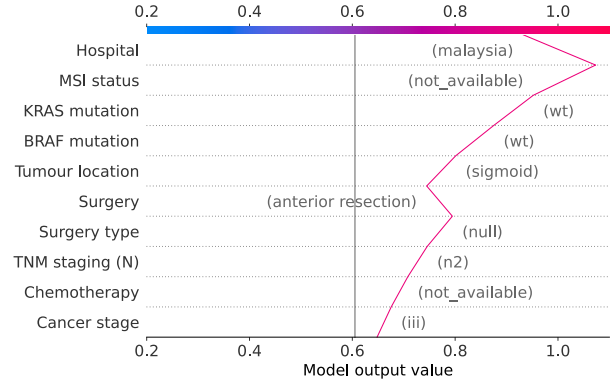
7.1. Individualised explanations

We obtain individualised explanation plots that are highly interpretable and offer insights into the decision process of our model. Figure 2 shows the importance scores of the top 10 most influential tabular features output for each patient, and outlines the decision path taken by the model in arriving at a prediction of recurrence or non-recurrence. We note that the explanations are generally aligned with some of the known prognostic factors that contribute to recurrence - for example genetic factors feature in the explanations, as well as tumour stage, location and the type of surgery performed.

We also note that the explanations have flagged out the MNAR data in some genetic testing features (KRAS, BRAF and MSI-status). Missing data in these features imputed as *'not_available'* were highlighted in the explanation as important factors. We can therefore infer that it is the "missingness" that is important and not the actual variable. That is,



(*a*) Explanation for a prediction of non-recurrence

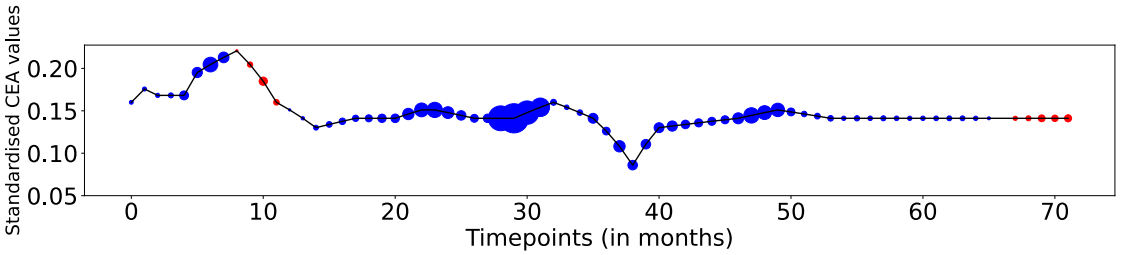


(*b*) Explanation for a prediction of recurrence

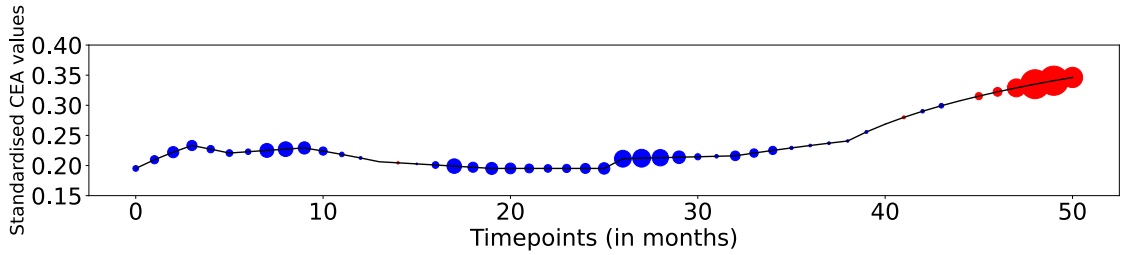
Figure 2: Decision plot showing how model arrived at its decision using tabular features. X-axis: model output (left: towards no recurrence, right: towards recurrence). Y-axis: Top-10 features in descending importance.

the true explanation variable for the recurrence outcome may not necessarily be the genetic test itself, but rather, the reasons that underlie the decision on whether or not to perform the test. We discuss MNAR data and its ramifications in Section 8.

We also output temporal explanations for the CEA measurement data in Figure 3. For a recurrence prediction (Figure 3(*b*)), we infer that the model considered the last 6 months of data as most informative in the recurrence prediction, and also that the risk increased with each consecutive month. This is understandable as the model had correctly picked up a trend between rising CEA levels and recurrence, and it lends assurance that the model makes use of reasonable features when performing the prediction. We also note that the model has learnt beyond simple associations between increasing CEA and recurrence risk: in Figure 3(*a*) the model disregards fluctuations



(a) Explanation for a prediction of non-recurrence



(b) Explanation for a prediction of recurrence

Figure 3: Normalised CEA values over time with SHAP values overlaid as coloured dots. Red/Blue signifies decision inclination towards recurrence/no recurrence while size of dots reflect decision magnitude.

and small rising trends, giving greater weight to a prediction of non-recurrence.

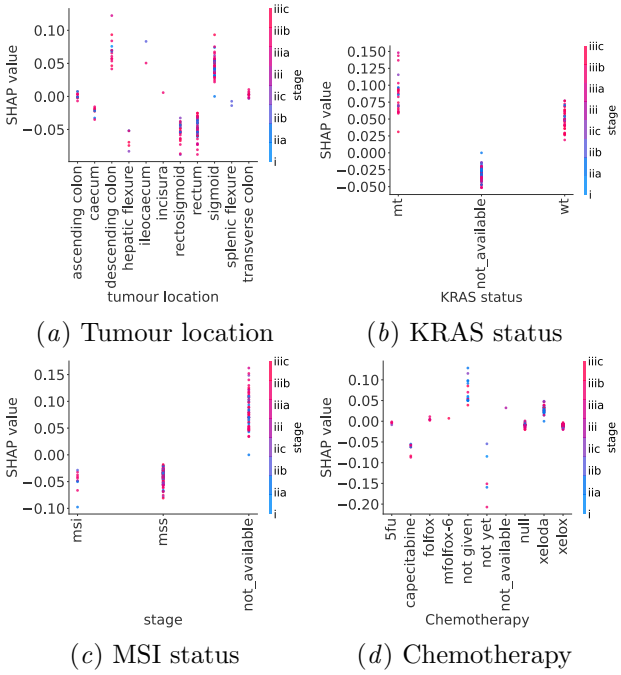
7.2. Understanding global associations

To elucidate potential associations between features and the recurrence outcome, we output SHAP global explanations in the form of dependency plots (Figure 4). We present and discuss the factors that we pre-identified as important in individual patients (Section 7.1), since factors important at an individual level are also likely to be important globally. We note that the complex interactions between factors may not be fully captured by our dataset.

Figure 4(a) shows that a primary tumour that originated from the descending colon and sigmoid was more associated with recurrence, and conversely one that originates within the rectosigmoid and rectum was less associated, irregardless of stage. This points to a possible difference in post-operative recurrence risk depending on anatomical location. Rectal cancer is known to differ from cancer originating from the sigmoid region in terms of genetics and biological characteristics (Wang et al., 2020; Frattini et al., 2004), thus it is not surprising that our study has elucidated a potential link between the tumour location and recurrence. Nonetheless, we note that our study is not powered to determine whether the difference is due to inherent biological variation between the two cancer types, or to differences in patient management.

Figure 4(b) highlights that a mutated ('mt') KRAS gene is more likely to be associated with recurrence, a finding that is supported by numerous studies (Kemeny et al., 2014; Margonis et al., 2016). Interestingly, our results seem to suggest that KRAS typing is usually only performed for tumours at higher stages, which could explain the observation that a *'not_available'* status is commonly linked to a lower risk of recurrence. On the other hand MSI status did not appear to be confounded by stage (Figure 4(c)).

In Figure 4(d), an interesting observation was that post-surgical patients who were not followed-up with chemotherapy treatment ('*not given*') had tumours that tended to be of a lower stage, yet also a higher tendency towards recurrence. It is tempting to apply a causal interpretation in this context, to deduce that a low stage tumour could be a possible reason for the decision to not provide chemotherapy, and such an act could have resulted in a higher risk of recurrence. Nevertheless we concede that the true situation was likely to consist of a more complex interplay of factors giving rise to the situation we observe, and that it is dangerous to come to conclusions without full understanding of the clinical context.



(*a*) Tumour location (*b*) KRAS status

(*c*) MSI status (*d*) Chemotherapy

Figure 4: Dependency plots for selected clinical variables versus tumour stage. Data points are SHAP values for individual patients (left axis) with information on tumour stage represented along a red-blue colour spectrum (right axis), for each category of the investigated clinical variable (bottom axis). SHAP values above/below 0 signify push towards recurrence/no recurrence, while value size is reflective of decision magnitude.

8. Reflections

Clinical implications: Our study demonstrates that longitudinal CEA readings combined with routinely collected clinical information is sufficient to strongly predict recurrence in CRC patients, using a multi-view deep learning framework. Our model `HTMV` achieves sensitivity, specificity and AUROC scores of 0.82, 0.96 and 0.95 respectively, exceeding the reported performance of both CEA-alone in the clinic (sensitivity ∼0.5, specificity ∼0.8), as well as that of several commercial diagnostic assays (AUROC of ColoPrint: 0.63, OncoDefender: 0.55). While the results are not directly comparable due to the absence of a standardised benchmarking dataset, the strong performance of our model is still heartening.

We have also utilised the SHAP framework to provide individualised and global explanations for the behaviour of our model, while accounting for the assumption of feature independence. Our explanations have highlighted possible associations between selected clinical features and the recurrence outcome, and pointed out potential biases in the dataset due to MNAR data. On the overall, our explanation module has achieved the objective of offering insights into the decision process of our model, and we hope that it has enhanced the trustworthiness of our algorithm.

Through our study conclusions, we hope to have contributed towards the development of clinically intelligible models in the healthcare space, without sacrificing on the performance and quality of the models.

MNAR data in healthcare: The explanations have highlighted that MNAR data were important features utilised by our model. There are two reasons why this is problematic: 1) The medical justification for not performing or reporting clinical test results is unrecorded in our training data, making further analyses to elucidate the true explanation variable difficult, and 2) MNAR data within the feature could introduce systematic biases in our training dataset, which may have an adverse impact on the performance of our model when applied in an alternative setting. For example, in a new situation where it was mandatory for all patients to be genetically tested, the model would be unable to leverage on the "missingness" status to make a prediction. We do not deny that due to the existence of such biases, our model's generalisation performance in the wild may be impacted. Nonetheless we would like to point out that this is a very common problem in real-world datasets, and we argue for the use of explanation modules such as ours to highlight where the problem area lies.

Next steps: We now identify several avenue for future studies. (i) Ours is a predictive model based on retrospective patient data; its generalisation performance is contingent on how accurate and representative the training data is. Prospective performance on data in the wild is a natural next step. (ii) As we have explained, it is difficult to make convincing arguments for variable associations without first solving the problem of MNAR data; this also has impact on how our model generalises. We intend to work with clinician stakeholders and data stewards to determine how to limit missing data and improve dataset quality. (iii) Lastly, it will increase clinical relevance if our model can proactively forecast recurrence early, say months before onset, and we intend to do so as part of our future work.

Acknowledgments

This research is supported by the Ministry of Education, Singapore (under grant WBS R-263-000-D64-114), and by A*STAR, CISCO Systems (USA) Pte. Ltd and National University of Singapore under its Cisco-NUS Accelerated Digital Economy Corporate Laboratory (Award I21001E0002). We also thank National Cancer Centre Singapore for providing the datasets used in this study.

References

Kjersti Aas, Martin Jullum, et al. Explaining individual predictions when features are dependent: More accurate approximations to Shapley values. *Artificial Intelligence*, 298:103502, 2021.

Martın Abadi, Ashish Agarwal, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2015. Software available from tensorflow.org.

Okechinyere J. Achilonu, June Fabian, et al. Predicting Colorectal Cancer Recurrence and Patient Survival Using Supervised Machine Learning Approach: A South African Population-Based Study. *Frontiers in Public Health*, 9:838, 2021.

R. Adam and E. Vinet. Regional treatment of metastasis: surgery of colorectal liver metastases. *Annals of Oncology*, 15:iv103–iv106, 2004.

A. Castells, X. Bessa, et al. Value of postoperative surveillance after radical surgery for colorectal cancer: results of a cohort study. *Diseases of the Colon and Rectum*, 41(6):714–723; discussion 723–724, 1998.

Kumardeep Chaudhary, Olivier B. Poirion, et al. Deep Learning-Based Multi-Omics Integration Robustly Predicts Survival in Liver Cancer. *Clinical Cancer Research: An Official Journal of the American Association for Cancer Research*, 24(6):1248–1259, 2018.

Shaika Chowdhury, Chenwei Zhang, et al. Mixed Pooling Multi-View Attention Autoencoder for Representation Learning in Healthcare. *arXiv:1910.06456 [cs, stat]*, 2019. arXiv: 1910.06456.

Maximilian Christ, Nils Braun, et al. Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package). *Neurocomputing*, 307:72–77, 2018.

Milo Frattini, Debora Balestra, et al. Different genetic features associated with colon and rectal carcinogenesis. *Clinical Cancer Research: An Official Journal of the American Association for Cancer Research*, 10(12 Pt 1):4015–4021, 2004.

Hiroshi Fukui, Tsubasa Hirakawa, et al. Attention Branch Network: Learning of Attention Mechanism for Visual Explanation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10697–10706, Long Beach, CA, USA, 2019. IEEE.

Oscar G. F. Geessink, Alexi Baidoshvili, et al. Computer aided quantification of intratumoral stroma yields an independent prognosticator in rectal cancer. *Cellular Oncology (Dordrecht)*, 42(3):331–341, 2019.

R. A. Graham, S. Wang, et al. Postsurgical surveillance of colon cancer: preliminary cost analysis of physician examination, carcinoembryonic antigen testing, chest x-ray, and colonoscopy. *Annals of Surgery*, 228(1):59–63, 1998.

Kaiming He, Xiangyu Zhang, et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, Santiago, Chile, 2015. IEEE.

Danliang Ho, Iain Bee Huat Tan, and Mehul Motani. Predictive models for colorectal cancer recurrence using multi-modal healthcare data. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 204–213. Association for Computing Machinery, New York, NY, USA, 2021.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-term Memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Ora Israel and Abraham Kuten. Early Detection of Cancer Recurrence: 18F-FDG PET/CT Can Make a Difference in Diagnosis and Patient Care. *Journal of Nuclear Medicine*, 48(1 suppl):28S–35S, 2007. Publisher: Society of Nuclear Medicine Section: PET/CT in Cancer Patient Management.

Sarthak Jain and Byron C. Wallace. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

Dan Jiang, Junhua Liao, et al. A machine learning-based prognostic predictor for stage III colon cancer. *Scientific Reports*, 10(1):10333, 2020. Number: 1 Publisher: Nature Publishing Group.

Nancy E. Kemeny, Joanne F. Chou, et al. KRAS Mutation Influences Recurrence Patterns in Patients Undergoing Hepatic Resection of Colorectal Metastases. *Cancer*, 120(24):3965–3971, 2014.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*, 2017. arXiv: 1412.6980.

Scott Kopetz, Josep Tabernero, et al. Genomic Classifier ColoPrint Predicts Recurrence in Stage II Colorectal Cancer Patients More Accurately Than Clinical Factors. *The Oncologist*, 20(2):127–133, 2015.

C. Lea, M. D. Flynn, et al. Temporal Convolutional Networks for Action Segmentation and Detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1003–1012, 2017. ISSN: 1063-6919.

Peter F Lenehan, Lisa A Boardman, et al. Generation and external validation of a tumor-derived 5-gene prognostic signature for recurrence of lymph node-negative, invasive colorectal carcinoma. *Cancer*, 118(21):5234–5244, 2012.

Gershon Y. Locker, Stanley Hamilton, et al. ASCO 2006 update of recommendations for the use of tumor markers in gastrointestinal cancer. *Journal of Clinical Oncology: Official Journal of the American Society of Clinical Oncology*, 24(33):5313–5327, 2006.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st Conference on Neural Information Processing Systems*, pages 4768–4777, 2017.

Georgios A. Margonis, Yuhree Kim, et al. Codon 13 KRAS mutation predicts patterns of recurrence in patients undergoing hepatectomy for colorectal liver metastases. *Cancer*, 122(17):2698–2707, 2016.

Athanasia Nikolaou, Pedro Antonio Gutiérrez, Antonio Durán, et al. Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm. *Climate Dynamics*, 44(7-8): 1919–1933, 2015.

Fabian Pedregosa, Gaël Varoquaux, et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.

Marco Tulio Ribeiro, Sameer Singh, et al. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *arXiv:1602.04938 [cs, stat]*, 2016. arXiv: 1602.04938.

Rohit Saluja, Avleen Malhi, et al. Towards a Rigorous Evaluation of Explainability for Multivariate Time Series. *arXiv:2104.04075 [cs]*, 2021. arXiv: 2104.04075.

Tomohisa Seki, Yoshimasa Kawazoe, et al. Machine learning-based prediction of in-hospital mortality using admission laboratory data: A retrospective, single-site study using electronic health record data. *PLoS ONE*, 16(2):e0246640, 2021.

Ramprasaath R. Selvaraju, Michael Cogswell, et al. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. ISSN: 2380-7504.

Bethany Shinkins, Brian D. Nicholson, et al. The diagnostic accuracy of a single CEA blood test in detecting colorectal cancer recurrence: Results from the FACS trial. *PLoS ONE*, 12(3), 2017.

Ole-Johan Skrede, Sepp De Raedt, et al. Deep learning for prediction of colorectal cancer outcome: a discovery and validation study. *The Lancet*, 395 (10221):350–360, 2020. Publisher: Elsevier.

Nitish Srivastava, Geoffrey Hinton, et al. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Shiliang Sun, Liang Mao, et al. *Multiview Machine Learning*. Springer Singapore, 2019.

Mukund Sundararajan, Ankur Taly, et al. Axiomatic Attribution for Deep Networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017. ISSN: 2640-3498.

Caspar G. Sørensen, William K. Karlsson, Hans-Christian Pommergaard, et al. The diagnostic accuracy of carcinoembryonic antigen to detect colorectal cancer recurrence – A systematic review. *International Journal of Surgery*, 25:134–144, 2016.

Wen-Chien Ting, Yen-Chiao Angel Lu, et al. Machine Learning in Prediction of Second Primary Cancer and Recurrence in Colorectal Cancer. *International Journal of Medical Sciences*, 17(3):280–291, 2020.

Stef van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software*, 45(1): 1–67, 2011. Number: 1.

Ashish Vaswani, Noam Shazeer, et al. Attention is all you need. In *Proceedings of the 31st Conference on Neural Information Processing Systems*, pages 5998–6008, 2017.

Phongtharin Vinayavekhin, Subhajit Chaudhury, Asim Munawar, et al. Focusing on What is Relevant: Time-Series Learning and Understanding using Attention. *arXiv:1806.08523 [cs]*, 2018. arXiv: 1806.08523.

Liming Wang, Yasumitsu Hirano, et al. Left colon as a novel high-risk factor for postoperative recurrence of stage II colon cancer. *World Journal of Surgical Oncology*, 18(1):54, 2020.

Ellery Wulczyn, David F. Steiner, et al. Interpretable survival prediction for colorectal cancer using deep learning. *npj Digital Medicine*, 4(1):1–13, 2021.

Yanbo Xu, Siddharth Biswal, et al. RAIM: Recurrent Attentive and Intensive Model of Multimodal Patient Monitoring Data. *arXiv:1807.08820 [cs, stat]*, 2018. arXiv: 1807.08820.

Patrick. E. Young, Craig M. Womeldorph, et al. Early Detection of Colorectal Cancer Recurrence in Patients Undergoing Surgery with Curative Intent: Current Status and Challenges. *Journal of Cancer*, 5(4):262–271, 2014.

Jing Zhao, Xijiong Xie, et al. Multi-view learning overview: Recent progress and new challenges. *Information Fusion*, 38:43–54, 2017.

Appendix A. Formulae of 6 extracted features for time-series clustering

We calculate the following 3 features that are indicative of the homogeneity of points within the segment:

Variance , which measures variability from the mean:

$$s^2 = \frac{1}{t_s - t_{s-1}} \sum_{i=t_{s-1}}^{t_s} (y_i - \bar{y_s})^2$$

Skewness , which refers to the degree of asymmetry from a typical gaussian distribution:

$$g_1 = \frac{\frac{1}{(t_s - t_{s-1})} \sum_{i=t_{s-1}}^{t_s} (y_i - \bar{y_s})^3}{s_s^3}$$

where S_s is the standard deviation of the segment.

Kurtosis , which measures the extent to which the tails of our distribution differs from a gaussian distribution:

$$g_2 = \frac{\frac{1}{(t_s - t_{s-1})} \sum_{i=t_{s-1}}^{t_s} (y_i - \bar{y_s})^4}{s_s^4} - 3$$

The next two features measure the linearity of the segment, namely:

Slope of linear regression over all points within the segment:

$$\hat{\beta} = \frac{\sum_{i=t_{s-1}}^{t_s} (i - \bar{t_s})(y_i - \bar{y_s})}{\sum_{i=t_{s-1}}^{t_s} (y_i - \bar{y_s})^2}$$

where the numerator is the covariance between the indexes t and time-series values y for each segment, and the denominator is the standard deviation of the segment.

Mean-squared error of a fitted linear curve to the segment:

$$\text{MSE} = \frac{1}{t_s - t_{s-1}} \sum_{i=t_{s-1}}^{t_s} (y_i - \hat{y}_i)^2$$

where $\hat{y}$ are the predicted values of y from a least-squares fit.

Lastly we calculated:

Autocorrelation coefficient , which measures the degree of correlation between the segment itself when shifted by some time-delay k:

$$\text{AC} = \frac{1}{t_s - t_{s-1}} \sum_{i=t_{s-1}}^{t_s} (y_i - \bar{y_s})(y_{i-k} - \bar{y_s})$$

Appendix B. Architectural details of baseline models

B.1. Shallow ML

We used the Python package *tsfresh* (Christ et al., 2018) to extract relevant and meaningful features from time-series data. We selected only features that were statistically significant after accounting for multiple hypothesis testing. The processed features were concatenated with tabular data and fed into the models as a single dataset.

We implemented off-the-shelf ML classifiers using *scikit-learn* (Pedregosa et al., 2011). Parameters were selected through extensive grid-search on the validation dataset. We investigated 5 models and report the best parameters:

- **Logistic regression** (`LR`), C=0.1 and l2 penalty
- **Support vector machine** (`SVM`) radial basis function (RBF) kernel, C=1, gamma="scale"
- **Gradient boosted tree** (`GB`) learning rate=0.2, max depth=2, max features=285, min samples at leaf node=8, num estimators=45
- **Random forest** (`RF`) criterion='entropy', max depth=4, max features=150, num estimators=46
- **Multi-layer perceptron** (`MLP`) with two dense layers (70 and 10 nodes), relu activation, dropout (0.3 and 0.15), optimizer=Adadelta

To handle class-imbalanced data, we utilized a weighted loss function, setting weights to the inverse of the corresponding class support.

B.2. Deep temporal networks

We created and trained neural network architectures in a similar approach described in Section 4, except that we replaced the Transformer subnetwork with the following networks also commonly used to analyse temporal datasets.

Long Short-Term Memory We employed a bi-layer LSTM (Hochreiter and Schmidhuber, 1997) network with hidden layer size 8, tanh activation, dropout and recurrent dropout rate set to 0.2, and initial learning rate set to 1e-3. We utilised weighted cross-entropy loss to coerce the model to learn the minority class better on an imbalanced dataset.

Temporal Convolutional Network Referencing architecture first described by Lea et al. (2017), we employed 1D-CNNs that apply convolutions across the temporal domain. We stacked 6 convolutional blocks, each consisting of alternating 1D-convolutional and dropout layer, with residual connection between inputs and outputs. We employed causal padding to train model on early inputs. Other model specifications are: dilation rate 2, filter size 2, hidden layer size 60, relu activations for all layers except for classification, He uniform weight initialisation, and dropout rate 0.1.

Similar to `HTMV`, we combine each of these subnetworks with an MLP trained to extract features from tabular data, to generate two models that we refer to as `LSTM-mlp` and `TCN-mlp` respectively. The training methodology follows that described in Section 4.3.