Integrating Rule-Based eGFR Labels with Expert GP Annotations: A Multi-Method Framework for CKD Classification

Ali Guran[1]✉, Avishek Siris[1], Gary K.L. Tam[1], James Chess[2], and Xianghua Xie[1]

[1] Department of Computer Science, Swansea University, Swansea, UK SA1 8EN
935538@swansea.ac.uk
[2] Wales Kidney Research Unit and Morriston Hospital, Swansea, UK SA6 6NL

Abstract. Chronic Kidney Disease (CKD) is a progressive condition that, without timely intervention, can lead to end-stage renal failure requiring dialysis or transplantation. Accurate multi-stage classification of longitudinal CKD data is critical for early referral and treatment. However, expert-labeled data is often scarce and costly. In practice, two types of datasets are common: a large, rule-based set labeled using eGFR, and a smaller, more reliable set annotated by GPs using broader clinical observations. In this study, we examine whether noisy but abundant eGFR labels and accurate GP labels provide complementary signals and whether combining them improves five-class CKD stage classification. We evaluate several strategies, including pre-training on eGFR data, fine-tuning on GP data, and hybrid approaches. We propose a fusion-based method that integrates latent representations from both datasets. Across various encoder architectures (LSTM, Bi-LSTM, Transformer, CNN+LSTM, CNN+Bi-LSTM, CNN+Transformer, TCN, TCN+LSTM, TCN+Bi-LSTM, TCN+Transformer), we test on the GP-labeled set. Our fusion method consistently outperforms baselines, supporting the hypothesis of complementary signals and demonstrating that fusion improves performance. This provides a practical solution for clinical settings with limited expert labels but abundant, noisy pseudo-labels.

Keywords: Chronic kidney disease · Classification

1 Introduction

Chronic Kidney Disease (CKD) is a globally prevalent, progressive condition that silently deteriorates renal function over time, often culminating in end-stage renal failure requiring dialysis or transplantation. With an estimated 10% of the world's population affected and risk factors such as diabetes and hypertension steadily rising, CKD has become one of the fastest-growing causes of morbidity and mortality [22]. Early identification and accurate staging of CKD are critical to effective management, as timely intervention can significantly slow disease progression and reduce complications. However, early stages of CKD are often asymptomatic, making timely diagnosis difficult in routine clinical care.

Machine learning (ML) and deep learning (DL) models hold promise for assisting in early CKD detection and staging, particularly through automated analysis of longitudinal patient data. Yet one major barrier to their deployment lies in the quality and availability of annotated data. Clinical data annotation is costly and time-consuming, particularly when it requires input from general practitioners (GPs) or nephrologists who consider a range of nuanced clinical factors. In contrast, estimated Glomerular Filtration Rate (eGFR)—a widely available biomarker—can be used to algorithmically label patient records based on established thresholds, enabling large-scale pseudo-labeled datasets [22]. These eGFR-based labels offer scalability but may not always align with expert assessments due to their dependence on single-variable cutoffs and disregard for other clinical indicators.

This divergence in label quality leads to a practical but underexplored challenge in CKD classification. Do these two sources provide complementary information, and can leveraging both improve classification performance beyond what either could achieve alone? If so, how can we effectively combine two distinct types of datasets—large, noisy eGFR-labeled records and smaller, high-fidelity GP-labeled samples? While eGFR data offers scale, it overlooks the broader clinical picture captured by GP annotations, which are more accurate but limited.

To investigate this, we present a hybrid learning framework that systematically explores how to integrate these disparate data sources. Our method begins by training a base encoder on the eGFR-labeled dataset to learn general CKD progression patterns. We then fine-tune this encoder on a smaller dataset where eGFR-derived labels are calculated from the GP cohort, serving as an intermediate adaptation phase. Separately, we train a second encoder exclusively on the limited GP-labeled data to model expert-level decision boundaries. The core of our approach lies in a fusion-based method that merges latent representations from both the eGFR-trained and GP-refined models, thereby enabling the unified model to benefit from both the breadth of the pseudo-labeled data and the depth of the expert annotations.

We benchmark this method against several alternatives, including standalone models trained only on eGFR or GP data, as well as sequential pre-train–then–fine-tune setups. Experiments are conducted across three sets: a large eGFR-labeled set, a GP cohort with rule-derived eGFR labels, and a small gold-standard GP-labeled set. Our results demonstrate that the fusion approach outperforms all baselines in terms of classification accuracy and stage-level sensitivity. These findings confirm our central hypothesis: eGFR and GP-labeled data encode complementary signals that, when properly integrated, lead to more robust and clinically meaningful CKD classification.

Our contributions include:

- We explore a new challenge in CKD classification: whether complementary information exists between noisy eGFR labels and high-quality GP annotations, and how to effectively combine them to improve performance.
- We systematically evaluate various common strategies and techniques, including pre-training on eGFR, fine-tuning on GP data, hybrid approaches, and various backbone architectures.

- We propose a fusion method that integrates latent features from both eGFR and GP models, showing that complementary information exists for the first time, and it consistently improves baseline performance.

Our solution has the potential to be applied to other clinical scenarios that face similar challenges, such as hypertension classification using predefined blood pressure thresholds [13] and diabetes diagnosis based on HbA1c levels [2]. These scenarios, like CKD, involve abundant rule-based pseudo-labels but limited expert data, and our approach may improve prediction and generalization in such settings.

2 Related Works

Several studies have investigated machine learning (ML) and deep learning (DL) techniques for chronic kidney disease (CKD) classification, utilizing various dataset types and classification approaches. The majority of these studies rely on cross-sectional data, offering a snapshot of patient health at a specific point in time. Some research, however, has focused on using time-series data to better understand CKD progression. CKD classification methods are typically divided into binary classification (distinguishing CKD from non-CKD) and multi-class classification (staging the progression of CKD). While significant advancements have been made in CKD prediction, challenges such as data scarcity, inconsistent labeling, and limited model interpretability continue to hinder progress.

Many studies have focused on cross-sectional data for CKD classification due to its accessibility and ease of processing. Saha et al. employed machine learning models such as Logistic Regression, Decision Tree, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Artificial Neural Networks (ANN) for binary CKD classification, achieving high predictive performance [18]. Similarly, Ghosh et al. explored models like Logistic Regression, Random Forest, Decision Tree, XGBoost, and Naïve Bayes, with XGBoost yielding the highest accuracy in CKD detection [5]. Jaddoa et al. developed a KNN and SVM-based approach, incorporating missing value imputation to enhance CKD classification accuracy [11]. Gurusamy et al. compared several ML classifiers and found that tree-based models outperformed traditional algorithms in CKD detection [10].

Some researchers have expanded ML-based CKD classification beyond binary classification to multi-class classification, aiming to predict different stages of CKD. Debal et al. proposed a machine learning pipeline for both binary and five-class CKD classification using data from St. Paulo's Hospital in Ethiopia [3]. They applied feature selection techniques and evaluated models like Random Forest, SVM, and Decision Tree, achieving strong results, particularly with Random Forest. In contrast, Ilyas et al. used a cross-sectional UCI dataset to derive CKD stages based on eGFR and compared the performance of J48 and Random Forest classifiers [9]. Their model showed high accuracy in multi-stage CKD classification, demonstrating the potential of rule-based label derivation. However, both studies focused on cross-sectional data, rather than time-series data. Critically, cross-sectional data, often representing tests conducted on a single day, only provides a snapshot and may not accurately capture chronic kidney status due to potential fluctuations. A truly robust CKD classification requires evidence of persistent kidney damage or reduced function, typically assessed by considering trends in test results over a 90-day period [22].

Recently, [7] conducted the first study to perform multi-stage classification across all five CKD stages using multivariate longitudinal data. Utilizing data from the UK's SAIL Databank, the authors applied LSTM and Bi-LSTM models. While they adhered to KDIGO guidelines by labeling patient stages based on eGFR trends over the preceding 90 days, a potential limitation is that relying primarily on eGFR for labeling may overlook other important diagnostic markers. Notably, their direct comparison demonstrated that these longitudinal models significantly outperformed traditional cross-sectional techniques (such as Random Forest, SVM, Decision Tree, and Logistic Regression) applied to the same cohort, with the Bi-LSTM achieving the highest accuracy and better capturing the temporal dynamics of the disease. Despite certain limitations related to the dataset and methodology, this study highlights the critical importance of using longitudinal patient data, rather than relying on cross-sectional snapshots, to achieve accurate and reliable multi-stage CKD classification.

While significant progress has been made in CKD classification, challenges such as data scarcity and label quality remain. Some studies rely on costly, expert-labeled datasets, while others use noisy, rule-based eGFR labels. To address this, we explore whether complementary information exists between noisy eGFR labels and high-quality GP annotations and how to combine them for improved performance. Our framework employs a gating mechanism to reconcile eGFR-based encoders with expert GP data. This mechanism mirrors how general practitioners integrate eGFR with other clinical markers, allowing the model to leverage both abundant eGFR data and precise GP annotations. By combining these sources, we improve CKD staging accuracy, particularly in settings with limited high-quality clinical labels. This approach offers a scalable, clinically viable solution for improving CKD detection and progression analysis, overcoming the limitations of data and label inconsistencies.

3 Dataset

We utilize two distinct datasets: the Welsh Results Reporting Service (WRRS) [20] and the Welsh Longitudinal General Practice Dataset (WLGP) [19]. Both datasets are available through the Secure Anonymised Information Linkage (SAIL) Databank [4,12,15–17]. The SAIL Databank serves as a secure and trusted platform for accessing a broad spectrum of anonymized health and administrative data, supporting research while safeguarding privacy and confidentiality. It facilitates the linkage of various datasets, allowing researchers to analyze and combine information from WRRS, WLGP, and other sources to drive healthcare research and enhance public health outcomes in Wales.

The Welsh Results Reporting Service (WRRS) is a national digital system that enables healthcare professionals across Wales to access and manage pathology results, regardless of where tests are conducted. By linking data from all Welsh health boards, WRRS reduces duplicate testing, improves patient safety, and streamlines care. It works alongside the Welsh Clinical Portal (WCP), which consolidates patient records into a single, accessible platform. Together, WRRS and WCP enhance the efficiency and accessibility of health information across NHS Wales. The Welsh Longitudinal General Practice Dataset (WLGP) captures detailed GP records for 86% of the Welsh population, including symptoms, diagnoses, treatments, prescriptions, and referrals. Data is mostly entered by clinicians, with test results transferred from secondary care. Using anonymized identifiers, WLGP can be linked with other datasets, supporting research, healthcare planning, and improved patient outcomes in Wales.

Data Processing We construct two primary datasets for our analysis: one labeled using eGFR-based rules (**D1**) and another labeled by general practitioners (**D2**). Both datasets are multivariate and longitudinal, containing 14 attributes, including patient *age*, *gender*, and a range of blood test results: *red blood cell count*, *creatinine*, *alanine transaminase*, *alkaline phosphatase*, *albumin*, *white blood cell count*, *sodium*, *mean cell volume*, *potassium*, *mean cell haemoglobin*, *globulin*, and *total protein.* To examine the alignment and divergence between rule-based and expert-labeled approaches, we derive a third dataset, **D2_E**, by applying the same eGFR-based labeling rules from **D1** to the patient records in **D2**. This derived dataset is central to our analysis, as it enables direct comparison of how identical patient trajectories are labeled by automated rules versus clinical judgment. **D2_E** forms the basis for learning to reconcile discrepancies between label sources.

Due to variability in clinical testing schedules—such as inconsistent intervals and missing values—we standardized time spans across patients. For each patient, we defined the observation window as follows: the *start date* was the latest among all the earliest test dates across the 14 attributes, and the *end date* was the earliest among all the latest test dates. This ensures that all selected patients have complete records across all attributes for the duration of the selected period. We then filtered patients to those with at least two years of continuous data within this unified window. Missing values were imputed using linear interpolation, and the data were uniformly sampled at 10-day intervals to create fixed-length, temporally aligned sequences. These preprocessing steps were applied to both **D1** and **D2**. Following this process, approximately 10,600 patients were included in the eGFR-labeled dataset (**D1**) and 1,700 in the GP-labeled dataset (**D2**). We only mention approximate numbers here in accordance with the SAIL output disclosure policy [21].

To summarize, we explore three datasets:

- **D1:** A large dataset labeled using eGFR,
- **D2:** A smaller, GP-labeled dataset (considered the most reliable),
- **D2_E:** A smaller dataset derived from **D2**, but with calculated eGFR labels.

Table 1. Overview of datasets and their corresponding data cohort and label

Dataset	eGFR Cohort Data	GP Cohort Data	eGFR Label	GP Label
D1	✓		✓	
D2		✓		✓
D2_E		✓	✓	

While the labels provided by general practitioners are used for the GP-labeled dataset (**D2**), the eGFR values are used for labeling both the eGFR-based dataset (**D1**) and the derived dataset (**D2_E**). Table 1 shows the underlying cohort data and their corresponding labels. We also differentiate the term 'Cohort Data' to specifically refer to the data itself, excluding the labels. Further, eGFR Cohort Data and GP Cohort Data do not overlap. These three datasets were split into training and test sets, with 80% used for training and 20% reserved for testing.

4 Methodology

Our research investigates the best approach for combining complementary data sources to improve CKD classification. **D1** provides broad coverage but contains noisier eGFR labels, while **D2** offers accurate yet limited expert annotations, and **D2_E** allows for direct exploration of the relationship between eGFR and expert labels. We aim to leverage both the abundant but noisy eGFR-labeled data and the small, clinically precise GP-labeled dataset to enhance classification accuracy. Figure 1 illustrates the six methods (Methods 1–6) we explore, which all utilize these three datasets: **D1**, **D2**, and **D2_E**. Each method applies a distinct approach to training and fine-tuning encoders, culminating in a fusion method that combines the strengths of all models.

4.1 Methods

We consider the following four method variants.

Method 1 consists of two key steps: An encoder model (Encoder 1) is initially trained using D1 (the large eGFR-labeled dataset). This step aims to capture broad CKD-related patterns under the assumption that despite potential label noise, the large scale of D1 helps the encoder learn generalizable representations of CKD progression. Next, the same encoder (Encoder 1) is fine-tuned on D2_E, the small set of GP cohort data labeled with calculated eGFR. Although D2_E is smaller, it further refines the encoder's feature representations to better reflect the GP cohort data while still relying on eGFR-derived labels. These two steps, produces a model that has been exposed to feature learning from a large volume of rule-based eGFR-only labeled data, as well the small of GP cohort data. As the model have seen both set of data it may later be used to ease the learning of the relationships between eGFR and GP labels.

Method 2 focuses on building an alternative encoder exclusively using D2 (the small GP cohort data + GP labels). Here, an entirely new encoder (Encoder 2) is pre-trained on D2. Because D2 labels come from GP records (i.e., real expert

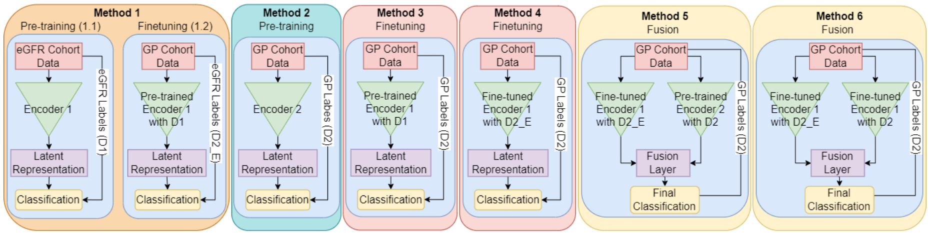

Fig. 1. Architecture

annotations), this encoder is expected to learn a more precise and clinically informed representation—albeit from a limited dataset size. The outcome of this method is a standalone GP-trained encoder; however, due to the small scale of D2, the trained encoder may lack the broader coverage of CKD variations captured in Method 1.

Method 3 and 4 While Method 3 further fine-tunes an encoder initially trained using Method 1.1 (i.e., pre-trained on eGFR cohort data with eGFR labels), Method 4 further fine-tunes an encoder initially trained using Method 1.2 (i.e., fine-tuned Method 1.1 on GP cohort data with eGFR labels) directly on D2 (GP cohort data + expert GP labels). These steps aim to align the model more closely with clinically validated annotations by transitioning from learning broad but potentially noisy patterns in the large eGFR-labeled dataset to refining those features using a smaller, high-quality GP-labeled dataset.

Method 5 and 6 (Fusion) are our final fusion methods, designed to combine the strengths of the previously trained encoders. While we use the latent representations from Method 1's fine-tuned encoder (trained on D1 and then D2_E), and Method 2's encoder (pre-trained on D2) for Method 5, we use the latent representations from Method 1's fine-tuned encoder (trained on D1 and then D2_E), and Method 4's encoder (trained on D1, fine-tuned on D2_E and then further fine-tuned on D2) for Method 6. These two encoders in Methods 5 and 6 can capture complementary information: one focuses on broad eGFR-based patterns and partial GP domain adaptation, while the other focuses on GP labels. In these methods, we use a gating mechanism to merge the latent representations generated from the two encoders. We use a gating mechanism to reflect how general practitioners often use eGFR as an initial reference point, then integrate it with other clinical markers. This approach allows the model to leverage both the abundance of eGFR data and the precision of GP annotations. The fused representations are then fed into a final classification head, which receives GP data (D2) for optimization. The intuition is that eGFR-based features, albeit imperfect, can still encode valuable CKD progression signals that, once directly fused with precise GP annotations, yield higher classification accuracy and better feature generalization.

Our gating-based fusion method combines representations from two pre-trained, frozen encoders using a learnable, element-wise weighted strategy. The embeddings are concatenated and passed through a gating network with sigmoid activation, which assigns weights to each encoder's contribution at the feature level. Only the gating layer and final classifier are trained (with Adam), enabling adaptive, efficient fusion while preserving the original encoder representations.

The above six-methods allow us to systematically evaluate how different training strategies—large-scale pre-training, smaller-scale fine-tuning, and fusion—can be combined to maximize the use of both eGFR and GP-labeled data. By contrasting Methods 1 through 4 against the final Methods 5 and 6 (Fusion), we show how leveraging both noisy but abundant eGFR labels and scarce expert GP labels can ultimately yield a more robust CKD classifier.

4.2 Encoders

We consider the following encoders in our experiments.

LSTM The encoder architecture comprises three stacked LSTM [8] layers. It takes a 14-feature input sequence, progressively reducing the hidden dimensions from 32 to 16, and then to 8. The final hidden state from the last LSTM layer is fed into a fully connected layer to produce classification scores over 5 classes.

Bi-LSTM A three-layer Bidirectional LSTM (Bi-LSTM) [6] model was also evaluated. It processes 14-feature input sequences with hidden sizes of 32, 16, and 8 per direction across the layers. Outputs are concatenated at each step (dimensions $64 \rightarrow 32 \rightarrow 16$). The final time step's hidden state (dimension 16) is passed through a fully connected layer to produce scores for 5 classes.

Transformer A Transformer [23] encoder was used for classification. Input sequences (14 features per time step) are linearly projected into 32-dimensional embeddings, with positional encodings added (max length 74). The model includes 2 encoder layers, each with 4 attention heads and a feed-forward network of size 64. The output from the final time step (dimension 32) is passed through a fully connected layer to produce scores for 5 classes.

TCN A Temporal Convolutional Network (TCN) [1] with 4 residual blocks was used for classification. Each block contains two 1D causal convolutions (kernel size 5, ReLU, spatial dropout 0.2) with exponentially increasing dilation factors (1, 2, 4, 8). The network processes 14 input features, with channel sizes of 64, 128, 128, and 64 across the blocks. After the TCN stack, Global Average Pooling is applied to the final block (64 channels). The pooled vector is passed through a fully connected layer (64 to 32 dimensions), followed by ReLU, dropout (0.2), and a final output layer for 5 classes.

CNN+LSTM A hybrid encoder combining a 1D CNN [14] and stacked LSTMs was developed. The 14-feature input passes through a convolutional layer (kernel size 3, padding 1, 64 channels), followed by BatchNorm and ReLU. The resulting 64-feature sequence is fed into three LSTM layers with hidden sizes of 32, 16, and 8. The final 8-dimensional output is passed through a fully connected layer to predict scores for 5 classes.

CNN+Bi-LSTM This encoder combines 1D CNN and Bidirectional LSTMs. The 14-feature input passes through a 1D convolution (kernel size 3, padding 1, 64 channels), followed by BatchNorm and ReLU. The output is processed by three stacked Bi-LSTM layers with hidden sizes of 32, 16, and 8 per direction.

Final forward and backward states from the third Bi-LSTM (dimension 16) are concatenated and fed into a fully connected layer to predict 5 classes.

CNN+Transformer A hybrid architecture combining CNN and Transformer encoders was implemented. The input sequence (14 features, length 74) passes through a 1D convolution (64 channels, kernel size 3, padding 1), followed by BatchNorm and ReLU. The output is projected to a 32-dimensional space, with positional encodings added for sequence length 37. The sequence is processed by 2 Transformer encoder layers (32 dimensions, 4 attention heads, 64-dimensional feed-forward network). The final 32-dimensional output is passed through a fully connected layer to predict 5 classes.

TCN+LSTM A sequential architecture combining TCN and LSTMs was implemented. The 14-dimensional input is processed by a TCN module with 4 residual blocks, using 1D causal convolutions (kernel size 5, dropout 0.2, dilation factors 1, 2, 4, 8). Channel sizes evolve as [64, 128, 128, 64]. After BatchNorm and ReLU, the 64-channel output is passed through three LSTM layers (hidden sizes 32, 16, and 8). The final 8-dimensional hidden state is passed through a fully connected layer to predict 5 classes.

TCN+Bi-LSTM A hybrid TCN-Bi-LSTM architecture was designed. The 14-dimensional input passes through a TCN with 4 residual blocks (kernel size 5, dilation factors 1, 2, 4, 8, dropout 0.2), with channels evolving as [64, 128, 128, 64]. After BatchNorm and ReLU, the output is passed through three Bi-LSTM layers (hidden sizes 32, 16, and 8 per direction). The final 16-dimensional concatenated hidden states are passed through a fully connected layer to predict 5 classes.

TCN+Transformer A hybrid TCN-Transformer architecture was built. The 14-dimensional input passes through a TCN with 4 residual blocks (kernel size 5, dropout 0.2, dilations [1, 2, 4, 8], channels [64, 128, 128, 64]). The output is normalized and activated, then projected to 32 dimensions for the Transformer. Positional encodings (32 dimensions, length 74) are added. The sequence is processed by 2 Transformer encoder layers (32-dimensional embeddings, 4 attention heads, feed-forward size 64). The final 32-dimensional output is passed through a fully connected layer to predict 5 classes.

5 Results

Table 2 summarizes the performance of ten encoders across seven training strategies (Method 1.1, Method 1.2, Method 2, Method 3, Method 4, Method 5, Method 6), evaluated on the D2 (GP-labeled) test datasets. The evaluation metrics include Accuracy, Precision, Recall, F1-score, and Specificity. Accuracy is the ratio of correctly predicted instances to all instances. Precision reflects how many of the predicted positives are actually positive. Recall (or sensitivity) shows how many actual positives are correctly identified. The F1-score balances precision and recall as their harmonic mean. Specificity indicates how well the model identifies actual negatives. We are not able to provide confusion matrices due to the SAIL Output policy [21].

Table 2. All methods were tested on GP-Labeled Dataset(D2). M: Method

Encoders	**Metrics**	**M 1.1**	**M 1.2**	**M 2**	**M 3**	**M 4**	**M 5**	**M 6**
LSTM	Accuracy	39.03	41.15	68.23	69.29	69.44	69.59	**69.89**
	Precision	38.17	31.80	69.81	70.71	70.45	70.58	**70.93**
	Recall	38.95	41.14	68.21	69.27	69.42	69.57	**69.88**
	F1-Score	34.07	33.52	67.36	68.43	68.79	68.95	**68.88**
	Specificity	84.74	85.29	92.06	92.32	92.36	92.40	**92.47**
Bi-LSTM	Accuracy	38.43	40.39	68.68	69.59	69.89	70.20	**70.50**
	Precision	37.63	30.96	70.21	70.95	70.88	70.99	**71.51**
	Recall	38.35	40.39	68.67	69.58	69.87	70.17	**70.48**
	F1-Score	33.47	32.79	67.83	68.71	69.24	69.44	**69.51**
	Specificity	84.59	85.10	92.17	92.40	92.47	92.55	**92.62**
Transformer	Accuracy	39.94	41.45	69.14	69.89	70.05	70.95	**71.10**
	Precision	58.97	52.10	70.56	71.19	70.99	71.78	**72.16**
	Recall	39.86	41.44	69.12	69.88	70.02	70.93	**71.09**
	F1-Score	35.32	33.99	68.24	68.98	69.37	70.22	**70.06**
	Specificity	84.97	85.36	92.28	92.47	92.51	92.74	**92.78**
TCN	Accuracy	40.09	41.45	68.99	69.89	70.20	70.65	**70.80**
	Precision	59.19	32.05	70.11	71.03	71.12	71.50	**72.19**
	Recall	40.01	41.44	68.97	69.88	70.18	70.63	**70.79**
	F1-Score	35.25	33.78	68.08	68.96	69.55	69.86	**69.96**
	Specificity	85.01	85.36	92.25	92.47	92.55	92.66	**92.70**
CNN+LSTM	Accuracy	39.64	41.30	68.99	70.05	70.20	70.95	**71.10**
	Precision	58.56	31.86	70.28	71.36	71.05	71.64	**72.15**
	Recall	39.56	41.29	68.97	70.03	70.18	70.93	**71.09**
	F1-Score	34.81	33.61	68.09	69.18	69.51	70.19	**70.11**
	Specificity	84.90	85.32	92.25	92.51	92.55	92.74	**92.78**
CNN+Bi-LSTM	Accuracy	39.79	41.60	69.29	70.20	70.35	71.26	**71.41**
	Precision	52.19	52.43	70.57	71.46	71.18	72.21	**72.48**
	Recall	39.71	41.60	69.27	70.18	70.33	71.23	**71.39**
	F1-Score	35.15	34.18	68.43	69.31	69.65	70.55	**70.48**
	Specificity	84.93	85.40	92.32	92.55	92.59	92.81	**92.85**
CNN+Transformer	Accuracy	40.54	41.91	69.44	70.50	70.65	71.41	**71.71**
	Precision	59.45	52.38	70.68	71.71	71.48	72.35	**72.83**
	Recall	40.47	41.90	69.42	70.48	70.63	71.39	**71.70**
	F1-Score	35.81	34.31	68.56	69.59	69.95	70.73	**70.88**
	Specificity	85.12	85.48	92.36	92.62	92.66	92.85	**92.93**
TCN+LSTM	Accuracy	40.54	41.75	69.29	70.20	70.50	71.41	**71.56**
	Precision	59.76	32.52	70.35	71.31	71.31	72.35	**72.66**
	Recall	40.47	41.75	69.27	70.18	70.48	71.39	**71.54**
	F1-Score	35.75	34.14	68.35	69.25	69.80	70.75	**70.73**
	Specificity	85.12	85.44	92.32	92.55	92.62	92.85	**92.89**
TCN+Bi-LSTM	Accuracy	40.85	41.91	69.59	70.65	70.80	72.16	**72.31**
	Precision	59.96	52.55	70.64	71.69	71.59	73.00	**73.51**
	Recall	40.77	41.90	69.58	70.64	70.78	72.14	**72.30**
	F1-Score	36.23	34.47	68.69	69.73	70.09	71.48	**71.46**
	Specificity	85.20	85.48	92.40	92.66	92.70	93.04	**93.08**
TCN+Transformer	Accuracy	41.15	42.21	70.20	71.10	71.26	73.07	**73.22**
	Precision	60.23	52.66	71.21	72.24	72.02	73.87	**74.38**
	Recall	41.07	42.20	70.18	71.09	71.23	73.05	**73.21**
	F1-Score	36.76	34.73	69.36	70.21	70.57	72.34	**72.44**
	Specificity	85.27	85.55	92.55	92.78	92.81	92.27	**93.31**

The results show that Method 6 consistently outperformed the other methods across all evaluated encoder architectures. A fusion of Method 1.2 and Method 4, Method 6 excelled, particularly in Accuracy and F1-Score. For example, with the TCN+Transformer encoder, Method 1.2 achieved 42.21% Accuracy and 34.73% F1-Score, while Method 4 reached 71.26% Accuracy and 70.57% F1-Score. Method 6, using the same encoder, surpassed both, achieving 73.22% Accuracy and 72.44% F1-Score, highlighting the effectiveness of the fusion approach. Overall, Method 5 ranked second, while Methods 1.1, 1.2, 2, 3 and 4 showed lower performance across the board.

Evaluating performance across different encoder architectures, especially with the best-performing Method 6, shows that models incorporating TCN and Transformer components generally yield superior results. The top-performing model was the TCN+Transformer encoder combined with Method 6, achieving the highest scores across multiple metrics. Other hybrid models, such as TCN+Bi-LSTM (72.31% with Method 6) and CNN+Transformer (71.71% Accuracy with Method 6), also showed strong performance with Method 6. In contrast, the basic LSTM encoder demonstrated more modest results (69.89% with Method 6) compared to the advanced encoders.

In summary, the findings demonstrate that Method 6, a fusion of Method 1.2 and Method 4, significantly improves CKD classification performance. Additionally, advanced sequential data processing architectures, especially those incorporating TCN and Transformer mechanisms like the TCN+Transformer encoder, deliver the best results for this longitudinal classification task.

6 Discussion

This study was conducted to address the following research questions:
RQ1: *Do the two data sources provide complementary information, and can leveraging both improve classification performance beyond what either dataset can achieve alone?*
D1: As shown in the previous results, we observed consistent improvements when combining these datasets (e.g., Method 5 and Method 6). Our hypothesis is confirmed: there is complementary information between the eGFR-rule-based-labeled dataset and the GP-annotated dataset, despite differences in their labeling approaches. The large but noisy eGFR-rule-based dataset ($D1$) provides valuable progression patterns embedded in the features, which support accurate predictions for the GP-annotated dataset ($D2$).

RQ2: *If so, how can we effectively combine the two distinct types of datasets—large, noisy eGFR-labeled records and smaller, high-fidelity GP-labeled samples?*
D2: Our results indicate that combining the two datasets ($D1$, $D2$) generally yields positive outcomes (Method 5, 6). Contrary to traditional transfer learning approaches (pre-training on a large pseudo dataset and fine-tuning on an expert dataset, as in Method 3 and 4), properly fusing the learned features from both encoders (Method 6) results in a more significant boost across all metrics. This validates our approach of using a gating mechanism to mimic how GPs rely on

eGFR labels as an initial reference before incorporating other biomarkers for the final GP-expert label. These findings highlight that a better understanding of the underlying data distribution and feature alignment in the latent space is crucial for effectively fusing and leveraging the complementary signals between the two datasets and labels. The use of encoders that better fit the longitudinal nature of the data is also critical for enhancing model performance. By leveraging architectures like TCNs and Transformers, which are designed to capture temporal dependencies, we can more effectively model the progression of CKD over time and improve classification accuracy.

7 Conclusion

This study investigates how to leverage large, noisy eGFR-labeled data alongside a small, accurate GP-labeled dataset to improve five-class CKD classification. We proposed six methods: three using eGFR-based pre-training and fine-tuning, one using only GP-labeled data, and two fusion approaches. Experiments across multiple datasets showed that each method offered unique strengths in CKD classification.

Results show that while eGFR-based labels are scalable, their noise limits performance on clinically validated data. In contrast, GP-labeled training offers higher precision but suffers from limited size. Combining both in our fusion methods (Method 5 and 6) significantly improved accuracy on GP-labeled tests while preserving eGFR dataset coverage, highlighting the benefit of integrating pseudo-labels with expert annotations.

Our method enables scalable CKD model development without sacrificing clinical precision, especially when expert labels are limited. Future work could explore more advanced fusion techniques, domain adaptation, or incorporate factors like comorbidities and lifestyle. Overall, our approach provides a flexible, data-efficient path to earlier and more accurate CKD detection.

Acknowledgements Ali Guran was funded by the Turkish Ministry of National Education through the Postgraduate Study Abroad Program. Gary Tam received support from the CHERISH-DE Centre via the International Mobility Award [62] and the Collaboration and Knowledge Exchange Support [92S] (EP/M022722/1). This work was also partially supported by the EPSRC National Edge AI Hub (EP/Y007697/1). This study makes use of anonymised data held in the Secure Anonymised Information Linkage (SAIL) Databank. We would like to acknowledge all the data providers who make anonymised data available for research. The responsibility for the interpretation of the information we supplied is the authors' alone. All research conducted has been completed under the permission and approval of the SAIL independent Information Governance Review Panel (IGRP) project number 1220. For Open Access, the author has applied a CC BY license to any Author Accepted Manuscript resulting from this submission.

References

1. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271 (2018)
2. Brož, J., Brabec, M., Krollová, P., Fačkovcová, L., Michalec, J.: Hba1c screening for the diagnosis of diabetes. Diabetologia **66**(8), 1576–1577 (2023)
3. Debal, D.A., Sitote, T.M.: Chronic kidney disease prediction using machine learning techniques. Journal of Big Data **9**(1), 109 (2022)
4. Ford, D.V., Jones, K.H., Verplancke, J.P., Lyons, R.A., John, G., Brown, G., Brooks, C.J., Thompson, S., Bodger, O., Couch, T., et al.: The sail databank: building a national architecture for e-health research and evaluation. BMC health services research **9**, 1–12 (2009)
5. Ghosh, S., Khandoker, A.: Investigation on explainable machine learning models to predict chronic kidney diseases. sci rep 14, 3687 (2024)
6. Graves, A., Schmidhuber, J.: Framewise phoneme classification with bidirectional lstm and other neural network architectures. Neural networks **18**(5-6), 602–610 (2005)
7. Guran, A., Tam, G.K., Chess, J., Xie, X.: Multi-stage chronic kidney disease classification on longitudinal data. In: International Conference on AI in Healthcare. pp. 120–133. Springer (2024)
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
9. Ilyas, H., Ali, S., Ponum, M., Hasan, O., Mahmood, M.T., Iftikhar, M., Malik, M.H.: Chronic kidney disease diagnosis using decision tree algorithms. BMC nephrology **22**(1), 273 (2021)
10. India, B.M.G., Vadlamudi, S., Guggella, C.R., Pinjari, H., Sanikommu, P.R.: Improving enhanced clinical decision making: Chronic kidney disease detection. In: 2023 International Conference on Artificial Intelligence for Innovations in Healthcare Industries (ICAIIHI). vol. 1, pp. 1–6. IEEE (2023)
11. Jaddoa, A.S.: Chronic kidney disease (ckd) diagnosis using machine learning methodology classifications. Iraqi Journal for Computers and Informatics **50**(2), 38–45 (2024)
12. Jones, K.H., Ford, D.V., Jones, C., Dsilva, R., Thompson, S., Brooks, C.J., Heaven, M.L., Thayer, D.S., McNerney, C.L., Lyons, R.A.: A case study of the secure anonymous information linkage (sail) gateway: a privacy-protecting remote access system for health-related research and evaluation. Journal of biomedical informatics **50**, 196–204 (2014)
13. Kim, H., Hwang, S., Lee, S., Kim, Y.: Classification and prediction on hypertension with blood pressure determinants in a deep learning algorithm. International Journal of Environmental Research and Public Health **19**(22), 15301 (2022)
14. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
15. Lyons, R.A., Jones, K.H., John, G., Brooks, C.J., Verplancke, J.P., Ford, D.V., Brown, G., Leake, K.: The sail databank: linking multiple health and social care datasets. BMC medical informatics and decision making **9**, 1–8 (2009)
16. Rodgers, S.E., Demmler, J.C., Dsilva, R., Lyons, R.A.: Protecting health data privacy while using residence-based environment and demographic data. Health & place **18**(2), 209–217 (2012)

17. Rodgers, S.E., Lyons, R.A., Dsilva, R., Jones, K.H., Brooks, C.J., Ford, D.V., John, G., Verplancke, J.P.: Residential anonymous linking fields (ralfs): a novel information infrastructure to study the interaction between the environment and individuals' health. Journal of Public Health **31**(4), 582–588 (2009)
18. Saha, B.P., Dash, S.R., Rout, N.K., Patnaik, A.P., Mishra, M.R.: Kidney disease prediction using ml techniques. In: 2024 International Conference on Emerging Systems and Intelligent Computing (ESIC). pp. 314–318. IEEE (2024)
19. SAIL: Welsh longitudinal general practice dataset (wlgp) - welsh primary care version 21.0.0 (2021)
20. SAIL: Welsh results reports service (wrrs) version 7.0.0 (2021)
21. SAIL Databank: Sail-pol-024 output review policy. PDF (2022), https://saildatabank.com/wp-content/uploads/2022/08/SAIL-POL-024-Output-Review-Policy-v1.2-3.pdf
22. Stevens, P.E., Ahmed, S.B., Carrero, J.J., Foster, B., Francis, A., Hall, R.K., Herrington, W.G., Hill, G., Inker, L.A., Kazancıoğlu, R., et al.: Kdigo 2024 clinical practice guideline for the evaluation and management of chronic kidney disease. Kidney international **105**(4), S117–S314 (2024)
23. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)