This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

Hospital Admission Location Prediction via Deep Interpretable Networks for the Year-round Improvement of Emergency Patient Care

Rasheed el-Bouri, David.W. Eyre, Peter Watkinson, Tingting Zhu* and David.A. Clifton*

Abstract*—*Objective: **This paper presents a deep learning method of predicting where in a hospital emergency patients will be admitted after being triaged in the Emergency Department (ED). Such a prediction will allow for the preparation of bed space in the hospital for timely care and admission of the patient as well as allocation of resource to the relevant departments, including during periods of increased demand arising from seasonal peaks in infections.** ***Methods:*** **The problem is posed as a multi-class classification into seven separate ward types. A novel deep learning training strategy was created that combines learning via curriculum and a multi-armed bandit to exploit this curriculum post-initial training.** ***Results:*** **We successfully predict the initial hospital admission location with area-under-receiver-operating-curve (AUROC) ranging between 0.60 to 0.78 for the individual wards and an overall maximum accuracy of 52% where chance corresponds to 14% for this seven-class setting. Our proposed network was able to interpret which features drove the predictions using a 'network saliency' term added to the network loss function.** ***Conclusion:*** **We have proven that prediction of location of admission in hospital for emergency patients is possible using information from triage in ED. We have also shown that there are certain tell-tale tests which indicate what space of the hospital a patient will use.** ***Significance:*** **It is hoped that this predictor will be of value to healthcare institutions by allowing for the planning of resource and bed space ahead of the need for it. This in turn should speed up the provision of care for the patient and allow flow of patients out of the ED thereby improving *patient flow* and the quality of care for the remaining patients within the ED.**

***Index Terms*—Machine learning algorithms, Multi-layer neural networks, Patient Flow, Hospitals.**

I. Introduction

DEEP neural networks (DNNs) have revolutionised the field of machine learning by providing a way to utilise very large datasets as well as large feature spaces to make meaningful predictions. State of the art performance has been achieved by DNNs in a wide range of tasks proving their efficacy as learning algorithms. Their strength in function approximation has not been overlooked by the medical community, with numerous publications exploiting them to make useful predictions for various healthcare scenarios [17, 30, 21].

One of the challenges of utilising DNNs is that they are non-convex optimisation problems meaning the best performance that the algorithm is capable of may not be achieved [8]. As a result, much work has been carried out in developing methods of presenting data to the network for training in a structured fashion [10]. This has since been called a curriculum and is widely used when training DNNs today.

The aim of this work is to utilise the concept of curriculum training to train a model that will predict where in a hospital a patient will be admitted based on very early information obtained in the ED from the triage nurse. We aim to show that the movement of patients from ED to one of seven different ward types in hospital is predictable. This would allow allocation of a bed and resources for the patient well ahead of admission to ensure that they receive care and treatment in as timely a fashion as possible. We also aim to demonstrate that this prediction can be done given data collected from a patient at point of entry to the ED department, which in turn will improve the flow of patients out of the ED and into the hospital. Difficulties in admitting patients to the optimal hospital ward are often most marked during periods of high demand, such as during peaks in seasonal infections including influenza. We therefore test the performance of our model through out the year.

In Section II we discuss the related work and in Section IV we discuss how a curriculum regularises the training of a DNN and how our algorithm is built. Then in Section VI we display the results of our algorithm and discuss these.

II. Related Work

In existing literature, there is currently much work published in the monitoring of patients in hospitals using machine learning techniques [12, 20]. However the application of machine learning to model patient flow is still a relatively new topic with a consequently limited literature.

Within this literature, prediction of admission to a particular ward based on measurements within hospital is a well explored area of research [18, 11, 1, 16]. Zhai *et al.* carried out work in predicting newly-hospitalised children who were likely to need transferral to the paediatric intensive care unit [23]. Logistic regression was used and achieved 89% accuracy. The model however only considered paediatrics, a subset of the total hospital population. While this is useful for the monitoring of the well-being of newly-hospitalised children it is not robust to be used as a general model for patient flow.

An investigation into the prediction of ward transition was carried out by Xu *et al.* in [35]. In this work, "alternating direction method of multipliers" (ADMM) was used in conjunction with discriminative learning of mutually correcting processes

R. el-Bouri, T. Zhu and D.A. Clifton are with the Department of Engineering Science, University of Oxford, Oxford, United Kindom. * - authors have equal contribution to the work.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

to learn and predict the destination of a ward transition. The model produced an overall next location prediction accuracy of 81% when considering all patients for all wards. It would seem that the model is powerful at predicting the transition process within the hospital, however it could also be argued that this is directly due to the data that have been used. In particular, they considered all patients within the hospital and did not discriminate between emergency and non-emergency patients. It is well known that good patient flow is significantly hindered by the ad-hoc introduction of emergency admissions into the hospital [26, 22]. The authors also use the MIMIC-II dataset [14] where the majority of the wards in consideration for transfer are ICU wards. This may not be useful for analysis of patient flow in the hospital as a whole. As a result, we will only consider patients who have been admitted in an emergency, we will consider all the wards within the hospital and we will aim to predict the initial point of entry.

In this work we choose to focus on the complex problem of predicting the outcome of the ED-inpatient interface (EDii). Staib *et al.* emphasise the importance of this interface by discussing how there is significant evidence to show that the delayed transfer of emergency patients to wards is associated with a 20-30% relative increase in inpatient mortality [34]. They also mention why this problem is difficult to predict. This is primarily due to the EDii being poorly defined in terms of clinical ownership, as well as the fact that the unscheduled nature of emergency admissions disrupts scheduled activity within the hospital, thereby slowing the movement of patients out of the ED. This can lead to patients being admitted to wards that are not ideal for their treatment in order to empty the ED, which can be hazardous [19]. By providing a prediction of the likely inpatient admission location, we seek to begin bridging the gap in patient flow between the ED and the inpatient wards.

Neural networks have primarily been used for the ward admission problem as binary classifiers. The majority of previous work using neural networks in this field predicts if a patient will or will not be admitted to a location within a hospital or to the hospital itself. Somoza *et al.* use a neural network to predict whether or not a patient presented to the ED of a psychiatric hospital will be admitted or not [3]. The model performs well using the neural network achieving a 91% accuracy. However this model is limited in its usefulness to clinicians on the ground. Knowing a patient will be admitted is useful for planning of overall numbers but greater granularity as to where they will be admitted is more useful for resource planning. As a result our problem will consider predicting the location of admission in the hospital.

In this work we utilise a curriculum in order to train our neural network. Curriculum Learning stems from the observation that children in schools learn by beginning with simple ideas and progressing on to more complex topics. By doing so they are able to understand fundamental principles on which they can build to learn more complex topics (which in themselves are usually simply superpositions of the fundamental principles). Curriculum Learning is the idea that neural networks may also benefit from this structured approach to learning. By presenting the network initially data that are 'easier' to optimise over, the optimisation surface (of network prediction error vs. network parameters) is more likely to be convex [10]. This has an analogy with numerical continuation methods, where a complex optimisation surface is decomposed into layers, beginning as a completely convex surface and gradually increasing in non-convexity [15]. In this paper we will exploit this methodology in order to train neural networks on noisy medical data. We will then compare this to normal batch methods of training networks and see the effect that the curriculum has on the prediction accuracy.

The use of non-stationary bandits in learning has also been explored in [31] where a curriculum is arranged and a bandit selects which batches to train a neural network on. The bandit is trained by measuring how a particular batch of data improves the performance of the network which in turn affects the probability of selecting that curriculum batch to train on. The better the performance, the more likely the bandit is to choose this batch of data again. The authors of [31] propose four different algorithms to select the next curriculum batch to train on. These are the use of a non-stationary bandit to select the next batch to train on, using linear regression and a windowed linear regression on the performance of the network to predict the batch most likely to provide the best performance after training, and using Thompson sampling to select the next batch for training. The authors found that the non-stationary bandit was the most effective method of choosing the next batch of data providing the best performance and faster training. While these approaches have an effective performance on the training problems presented in the work, the authors do not utilise the curriculum to guide their network weight space into the domain of a global minimum. Another work which uses a similar approach is that of [29] where a curriculum is also generated and a non-stationary bandit is used with the EXP3.S algorithm [7] to select the next curriculum batch to train on. However, once again without using the curriculum initially, this algorithm will not always provide a better or faster training of the network.

Aside from simply improving the accuracy of a model it is important, particularly when using deep models in the healthcare domain, to provide a level of interpretability to the decision making process. In [33], the authors emphasise the importance of understanding what in the input space has driven a decision in order to learn from the model, or to validate the classification. We again see this in a review of deep learning in healthcare by [32] where one of the fundamental challenges noted is interpretability of deep learning models and relating the decision made back to the input space. As a result we propose a 'saliency term' to see the most important features that contribute to predictions in our model.

III. Novelty

The novelties of this work are as follows: we have developed a novel strategy for the training of neural networks combining a curriculum training phase with a multi-armed bandit phase to maximise prediction performance on noisy biomedical data. This also incorporates a saliency layer before the inputs which allows interpretation of the importance of the input features. To

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

the best of the authors knowledge no other work has proposed the framework of predicting where in the hospital a patient from the ED will be admitted. This is also believed to be the first work to employ deep learning architectures in order to carry out hospital admission prediction.

IV. Methodology

A. Curriculum Learning

Due to the non-convex nature of optimising artificial neural networks (ANNs), a structured method of presenting data to the network via curriculum learning was introduced with the aim of reducing the likelihood of the weights being optimised into a local minimum [10]. There are similarities between curriculum learning and numerical continuation methods as pointed out in [10], where optimisation of a complex surface is achieved through first optimising over smoother more convex versions of the surface. Consider a family of cost functions $C_\lambda(\theta)$ such that C_0 is easy to optimise over (and which is likely to be more convex than other functions), $\lambda \in [0,1]$ is the ranking of "difficulty to optimise" and where C_1 is the actual cost function that is to be minimised. By optimising over the network parameters, θ, for C_0, as C_0 is simply a smoother version of C_1 we bring our parameters into the domain of a minimum of C_0 as well as C_1. We then gradually increase λ while keeping θ at the local minimum. This helps to avoid local minima which may be present in the more complex optimisation space. The aim therefore, is to create batches of data, Q, ranked according to λ (i.e., Q_λ with $\lambda = 0$ being the "easiest" batch of data to optimise progressing to the "hardest" as λ increases.) These batches are then presented to the network for training in order of increasing λ. Note that the batch $Q_{\lambda+\epsilon}$ will contain all of the data in Q_λ for $\epsilon > 0$, as an increment in λ represents the addition of more "complex" data to the previous batch.

With application to real data, we need to define "easiness" of fitting to the data. We define a sequence of batches of data $Q_\lambda(z)$ comprised of individual data entries, z, such that $\int Q_\lambda(z)dz = 1$ (i.e., our whole dataset). We also define $Q_\lambda(z) \propto W_\lambda(z)P(z)\ \forall z$, where $W_\lambda(z)$ is the weight assigned to example z at the point λ in the curriculum sequence and $P(z)$ is the training data ($W_\lambda(z)$ is 0 for "complex" data at low values of λ i.e, excluded in the "easy to optimise" batches). The "easiness" of the fit to data is described by:

$$H[Q_\lambda(z)] < H[Q_{\lambda+\epsilon}(z)] \qquad \forall\ \epsilon > 0 \tag{1}$$

where H is the entropy of data batch Q. The weights of the examples also increase with λ as:

$$W_{\lambda+\epsilon}(z) \geq W_\lambda(z) \qquad \forall\ z,\ \forall\ \epsilon > 0 \tag{2}$$

to balance training as the "less complex" data will have been presented to the network for training a greater number of times. This is because the first curriculum batch ('easiest') will also be a part of all the other curriculum batches, i.e, for N curriculum batches denoted by Q, $Q_0 \subset Q_1 \subset \dots Q_N$. Therefore the data in Q_0 is presented to the network a greater number of times and so the rest of the data must be weighted to account for this so that all data is presented an equal number of times.

In this work we define "complexity" of the data using the Mahalanobis distance in order to encode the notion of entropy. The Mahalanobis distance is a multi-dimensional generalisation of measuring the number of standard deviations that exist between a point P and the mean, μ, of a probability density function (p.d.f), D [4]. The larger the Mahalanobis distance the more unlikely the data entry is to belong to the distribution (and which is therefore of higher entropy). We therefore assume that our data belong to a single p.d.f, with mean μ and covariance $\mathbf{S}$. Due to the input features being of mixed data types, we encode our input features through a trained denoising autoencoder to gain a representation of the data in an embedded space before calculating the Mahalanobis distance. In using the Mahalanobis distance, our curriculum organises our training data such that we train according to the most similar samples first (the smaller number of samples of different classes in this batch increases the likeliness of finding a more global minimum, therefore making it "easier" to optimise over) before progressing on to the easier to differentiate between samples. This mirrors the approach that is used in the SVM in defining the separation boundary where data of differing classes are closest together.

B. Regularisation using a Mahalanobis Curriculum

We now postulate how the Mahalanobis curriculum may naturally regularise itself. Let $\mathcal{Z}$ be a training data set consisting of datapoints z_n where $z_n \in \mathcal{Z}$ and z_n consists of input features and a label such that $z_n = \{x_n, y_n\}$.

We also define the Mahalanobis distance as:

$$d_{m_n} = \left((\boldsymbol{x_n} - \boldsymbol{\mu})^T \boldsymbol{S^{-1}} (\boldsymbol{x_n} - \boldsymbol{\mu}) \right)^{\frac{1}{2}} \tag{3}$$

where $\boldsymbol{x_n}$ are the (continuous) input features of the datapoint, $\boldsymbol{\mu}$ is the vector of the mean value of each feature, and $\boldsymbol{S}$ is the covariance matrix.

Using this equation we can now create a vector, D_m, of distance of each datapoint from the mean of the assumed p.d.f of the dataset, where $\mathcal{X} \rightarrow D_m,\ \ \forall\ x_n \in \mathcal{X},\ \ \mathcal{X} \subset \mathcal{Z},\ \ \mathcal{Z} \subset \mathbb{R}$.

We now seek to create N batches of training data of increasing entropy of size k datapoints where $k = \frac{\mathbf{card}(D_m)}{N}$.

We then extract the indices of the lowest entropy features using the following formulation:

$$j_N = index \left(\bigcup_{i=1}^{i=Mk} \min \left((\dots (d_m \backslash d_{m_1}) \backslash d_{m_2}) \dots \backslash d_{m_i} \right) \right) \tag{4}$$

for $M = \{1, 2, ..., N\}$, and d_{m_b} is the b_{th} smallest element of the set D_m. We are then able to construct the N curriculum batches $B_N = \mathcal{Z}\{j_N\}$ and their corresponding outputs, $O_N = \mathcal{Y}\{j_N\}$. The training proceeds by presenting the batches in B for the smallest N first and then gradually increasing N.

Consider a typical cost function used for backpropagation: $\frac{1}{N}\sum_{n=1}^{N}(\hat{y_n} - y_n)^2$, which can be re-written as $\frac{1}{N}\sum_{n=1}^{N}(Wx_n - y_n)^2$ where W is an operator equivalent

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

to multiplication by the weights of the previous layers of a deep neural network. We are able to do this in this case as we activate the nodes of our network with 'relu' activations which is simply a piecewise linear operator.

Using the definition of the Mahalanobis distance as shown in Equation 3, if we consider x_n to be a random variable, we see for normally distributed data $x_n \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{S})$ and $\boldsymbol{x_n} \rightarrow \boldsymbol{\mu} + \sqrt{\boldsymbol{S}} d_m$. For ease of notation, we assume that all input features are orthonormal, i.e, $\boldsymbol{S}$ is a diagonal matrix. Therefore we see that $\boldsymbol{x_n} = \left(d_m^2.(\boldsymbol{SI})\right)^{\frac{1}{2}} + \mu$, where $\boldsymbol{I}$ is the identity matrix and which we substitute back into our expression for MSE, which expands to the following expression:

$$MSE = \frac{1}{N} \sum_{n=1}^{N} (W^2 \left(d_{m_n}^2(\mathbf{SI})\right) + 2W^2 \left(d_{m_n}^2(\mathbf{SI})\right)^{\frac{1}{2}} \mu - 2W \left(d_{m_n}^2(\mathbf{SI})\right)^{\frac{1}{2}} y_n + W^2\mu^2 - 2W\mu y_n + y_n^2)$$

When we are training with a curriculum, we train initially with low entropy data so that $d_{m_n} \rightarrow 0$: $MSE \rightarrow \frac{1}{N}\sum_{n=1}^{N} \left(W^2\mu^2 - 2W\mu y_n + y_n^2\right) = \frac{1}{N}\sum_{n=1}^{N} \left(W\mu - y_n\right)^2$. For very low entropy values we are simply calculating the mean squared error with respect to the mean of our assumed p.d.f.

Now we investigate as d_m becomes large: We assume that the first 3 terms in the expanded MSE equation will dominate the response due to the large value of d_m:

$$MSE \rightarrow \frac{2}{N} \sum_{n=1}^{N} \left[W \left(d_{m_n}^2.(\mathbf{SI})\right)^{\frac{1}{2}} \left(\frac{W}{2} \left(d_{m_n}^2.(\mathbf{SI})\right)^{\frac{1}{2}} + [W\mu - y_n] \right) \right]$$

Here there are two important things to notice: firstly the cost function now contains an additive loss term proportional to $||W||$. This means that in the case of overfitting where the magnitude of the weights increases dramatically, the error function will be penalised for this. This is artificially introduced using L1/L2 regularisation whereas here it naturally arises with data that is perceived to be of higher entropy. The next point to notice is that the difference between prediction and label is no longer squared meaning we have much more gradual learning with higher entropy data (which is positive as we don't want to learn the noise that is associated with these data).

By using a curriculum we initialise our function approximation using the mean of the data. This is advantageous as it greatly reduces the likelihood of our function approximation being skewed by outliers and possibly even erroneous data.

C. Multi-armed bandits

The curriculum is trained in a cyclical fashion which, as described previously, is beneficial for finding a local minimum near the global minimum. However after initial training there is no reason why this cyclical training should provide the best possible performance of the model. Given that we now have discrete batches of data created by the curriculum, we introduce a multi-armed bandit in order to choose the best batch to train the network on.

A multi-armed bandit is a method in which choices need to be made based on allocation of a finite resource, where the aim is to maximise the expected reward of allocation of the resource [13]. The probabilities of reward based on choice are only partially known at the time of allocation and the optimal choice to maximise reward becomes more clear as resource is spent. The multi-armed bandit is an example of an exploration vs. exploitation problem as is often framed within reinforcement learning problems. A hyperparameter that is manually chosen, ϵ, defines the rate with which exploration of the choices occurs (by choosing a batch at random) as opposed to exploiting the batch with the highest reward. Due to the non-convex nature of training an ANN, we can view the training of the ANN as a multi-armed bandit problem. For multi-class classification, certain classes are learned more rapidly depending on the data that has been presented to the network to train it. By using the concept of batches of data split according to their "easiness" as introduced by curriculum learning, we can treat this as a problem of choosing the right data to train the network on in order to maximise our reward which in this case is the general accuracy of the model in a multi-class classification.

Algorithm 1 The multi-armed bandit for training of the network after initially trained with a curriculum

procedure INITIALISATION
  rate of exploration = ϵ
  resource available = N_a
  Prob. curriculum batch gives max reward = P
  Num. training data batches = $N_{choices}$
  Count of number of times batch is chosen = K
  Batches by Mahalanobis distance = $c_{batches}$
  loop:
  for i in N_a **do**:
    if $\epsilon > u \sim U(0,1)$ **then**
      batch $= c_{batches}[int\,(u \sim U(0, N_{choices}))]$
    else
      batch $= c_{batches}[\arg\max(P)]$
    Train on batch and find accuracy on training set
    $i_0 \rightarrow A_T^0 = \sum_j^C (\delta_j)$
    $i_{1:N_a} \rightarrow A_T^i = \sum_j^C \left(\left(\delta_j^i - \delta_j^{i-1}\right)/\delta_j^{i-1}\right)$
    Test on validation set
    $A_v^i =$ overall accuracy on validation set
    reward $= A_T^i \times A_v^i$
    K[batch] = K[batch] $+ 1$
    $\alpha = 1/K$[batch]
    P[batch] $= P$[batch] $+ \alpha \times$ (reward$-P$[batch])

Algorithm 1 shows how the multi-armed bandit problem was applied for training. We begin by defining the exploration rate, ϵ, how many batches of data we have, $N_{choices}$, and how many attempts we have at training the network with the batches, N_a. We also initialise vectors of zeros of the same length as the number of training data batches, K and P.

For a value of $\epsilon = 0.1$, the bandit would explore (choose a different training data batch at random) 10% of the time. Otherwise the bandit will choose the training data batch that has the greatest probability of returning maximal reward.

Once the training data batch has been chosen we train

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

using these data. The reward is then calculated. For multi-class classification, we require a reward function that will improve the accuracy of prediction over all classes and not just the classes that are more prevalent in the data. We therefore define our reward function with respect to the learning rate of all the classes as well as the performance on the validation set to ensure that the model does not overfit.

$$R = \sum_{i=1}^{C} \frac{\delta_i^n - \delta_i^{n-1}}{\delta_i^{n-1}} \times A_v^n \tag{5}$$

where A_v^n is the validation set accuracy of the current training episode, δ is the accuracy of class i over the training set and n is the current training episode. By incorporating A_v^n, as soon as the model begins to overfit on the training data, reward due to the first term in Equation 5 will increase; however, any detriment to the general performance will be reflected by A_v^n which will prevent the reward increasing (i.e, a decrease in the accuracy over the validation set would lead to the sum of the learning gradients being multiplied by a small number thereby reducing the reward).

D. Prediction Interpretation

After training the model it is useful to understand from the clinical perspective why the model has made its predictions and why errors arise. We investigate this by modifying the architecture of our model slightly. We add a layer of weights to the input space that are multiplied element wise by the inputs changing the function approximator from $f(y \mid x; \theta)$ to $f(y \mid w_{in} \odot x; \theta)$. Having multiplied the inputs, x, by the weights w_{in} we then pass the weights through the *softmax* function to find the relative importance of each feature to the prediction and then add the entropy of this output to the cost function. We therefore change our cost-function so that it now becomes:

$$\mathcal{L}(\theta) = -\left(g(w_{in}) \log(g(w_{in})) + \sum_j (y_j \log(\hat{y_j})) \right) \tag{6}$$

where g implies the *softmax*, w_{in} are the pre-multiplying weights of the inputs, y_j is the real one-hot label of the prediction, $\hat{y_j}$ is the models predicted distribution over the classes and j is the data point. Using this loss we then use backpropagation as usual and update both θ, the network weights, and w_{in}. The effect of this function is to encourage sparsity in the inputs while maintaining the objective of classifying the patients. This will allow us to see the most important features for this prediction problem. We train until we achieve the same accuracy as was achieved previously with the knowledge that we have achieved the maximum performance possible with as sparse a feature space as possible.

V. Dataset

In this study we considered the patient data collected in the electronic health records (EHR) of Oxford University Hospitals (OUH), between January 2013 and April 2017. De-identified patient data were obtained from the Infections in Oxfordshire Research Database (IORD) which has generic Research Ethics Committee, Health Research Authority and Confidentiality Advisory Group approvals (14/SC/1069, ECC5-017(A)/2009). The EHR stores all digitally recorded data on an incoming patient. This includes administrative (e.g. date and time of arrival), demographic (e.g. age, gender and so on), as well as physiological and medical information (e.g. vital sign measurements and medical tests ordered during the patient's visit). Any historical data stored about the patient will also be available in the EHR upon their next arrival to the ED. To avoid learning from events where patients are admitted to wards not appropriate for their primary diagnosis and treatment, i.e. wards from another medical specialty, we exclude these admissions from the dataset. We filter patients according to whether or not their primary diagnosis code for the visit clearly corresponds to an appropriate label for their treatment (i.e., which ward they are admitted to). Those admitted to a ward obviously not appropriate for their treatment were disregarded. The features used for prediction can be found in Tables I and II. Only patients who were admitted in an emergency and who had a full set of the features listed in the appendix were considered providing a dataset of 9324 patients. The full dataset contains data from 51,277 unique patients admitted to the OUH via the ED. Upon filtering to only include adults and inclusion of a full feature set, our dataset reduces to 9,324. As a result, we seek to initially keep all features to prove that the problem is predictable before looking in future work as to how to reduce the number of features we are dependent upon to maximise utility to the hospital. A training set of 60% of the dataset was used and was balanced (on the basis of admitted ward group) leaving 5327 patients for training on. The validation set was 20% of the dataset and testing was also 20% and the classes were kept in the same distribution as the original dataset.

To validate the efficacy of the methodology we implement the algorithm on another classification problem from the MIMIC-III dataset in the next section [27]. The patients for this dataset are also emergency patients only and have all features available. This provides us with a dataset of 8806 patients. These were split into the same train-validation-test proportions as before with only the training set being balanced as before. As MIMIC-III is an ICU focused dataset, replicating the experiment we have carried out with the OUH dataset is not possible. As a result we create a new problem of classifying the mortality of patients (binary classification) based on 11 features that are available early in the patient's admission. All features used are shown in Tables I, II and III.

VI. Results and Discussion

The OUH hospital in consideration has a total of 108 unique wards. To create a more meaningful and useful predictor, these were grouped by experienced clinicians working in the hospital into seven 'ward types' based on the type of patient that is admitted and the function of the ward. These are medical, cardiac, neurosurgical/neurology (neuro), trauma, ICU, surgical and general / obstetrics & gynaecology (general/O&G) ward types.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

Bacteriology test requested?	Biochemical tests requested?
Blood cultures requested	Blood gas test requested?
CT scan requested?	Cardiac enzyme test requested?
Clotting study requested	Blood cross-matching requested?
Diastolic blood pressure at entry	Dental investigation requested?
ECG requested?	Heart rate at entry
Haematology test requested?	Immunology test requested?
MRI scan requested?	Early warning score
Vital signs requested?	No tests requested
Orthopedic tests requested?	Other tests requested?
Pregnancy test requested?	Respiratory rate at entry
Systolic blood pressure at entry	SPO_2 at entry
Serology test requested?	Body temperature at entry
Toxicology test requested?	Ultrasound test requested?
Urine test requested?	X-ray scan requested?
Admission method	Admission source
Age	experiencing atrial fibrillation?
# historic diagnoses	Previous management
Previous admission to ED?	Ethnic category
Frequently admitted?	Gender
Dist. of address to hospital	No. Investigations requested
Previous ED visit days ago	Previous visit LOS
Mortality indicator severity score	Historic diagnosis codes
Previous specialty	

TABLE I: Table containing the patient specific features available at initial medical assessment that were used in all of the models.

Triaged in 1 hour?	Time of triage
Average day temp (degrees)	ED capacity
ED attendance ID	Hour admitted to ED
Month admitted to ED	# ED Attendees in last 12 hours
# ED Attendees in last 4 hours	# ED Attendees in last 8 hours
# ED Attendees in last hour	# Breaches last 12 hours
# Breaches last 4 hours	# Breaches last 8 hours
# Breaches last hour	# Major admissions last 12 hours
# Major admissions last 4 hours	# Major admissions last 8 hours
# Major admissions last hour	Has it rained today?
Hours of sunlight	Max. day temp (degrees)
Min. day temp (degrees)	Weekday [0-6]

TABLE II: Table containing the environmental/hospital features that were used in all of the models.

Admission type	Admission location
Insurance	Language
Religion	Marital Status
Ethnicity	Has previous chart events
Previous ICD9 code	First care unit
First ward ID	

TABLE III: Table containing the features used for mortality prediction on the MIMIC-III dataset. Features are defined in MIMIC-III documentation.

The aim of the algorithm is to classify the patient as being admitted to one of these seven ward types. Initially, a multiple logistic regression and an SVM were used for the task (trained using stochastic gradient descent). These however provided poor performance, with the prediction accuracy being 14% for both methods, close to that of chance given a seven class classification. We then implemented our curriculum training methodology on both simple classification models as is undertaken in [24], to determine whether or not the proposed curriculum learning could improve their performance. We found that a simple linear regression model had its classification accuracy unchanged with or without curriculum learning, whereas the SVM improved from 14% accuracy to an average of 17% accuracy when using the curriculum only, and to an average of 21% when the curriculum is combined with the proposed multi-armed bandit.

In Figure 1 we implement a feedforward neural network for the hospital admission location problem. Use of the feedforward network provides good performance for the multiclass classification for some classes but not for all as indicated in Table IV. The maximum accuracy achieved on the valdiation and held-out test sets was 39% over all classes. However it can also be seen from Figure 1 that the loss and accuracy plots are very noisy. The five different seeds all provide very different performances at the end of training with a difference of approximately 10% performance on the validation set as seen in the accuracy plot in Figure 1. The range of losses shown in the loss plot indicates to us that after training the five seeds have found different local minima within the weight space. This indicates that this is not a very stable place from which to launch a non-stationary bandit search of the weight space as for different seeds we will be starting our optimisation from different locations and our final performance will be dependent on the inital seed.

In Figure 2 we repeat the experiment however this time incorporating a curriculum into the training regime. Using a Mahalanobis based curriculum not only achieves a higher maximum accuracy overall (46% over all classes) than stochastic mini-batch training, but also smoothes out the accuracy and loss of the five seeds. As can be seen in Figure 2, The range between the best performing and worst performing seeds is much smaller. We also see in the loss plot that all seeds eventually converge to the same loss, indicating that due to the curriculum all of the seeds have converged to a very similar local minimum. This not only improves the performance for the whole classification but also improves the performance of the individual classes that did not perform well initially which can again be seen in Table IV. The losses and the accuracies being much smoother provides us with a stable basis to begin an exploration vs. exploitation approach to training the network.

The multi-armed bandit is then incorporated into Figure 3, showing how the bandit explores until it finds the best batches to train the network on given what has previously proven successful. We are able to exploit the batches of data in the curriculum to provide us with a better or equal performance to a network trained only using a curriculum. We see in Figure 3 that the average accuracy initially decreases due to the exploration that is required and eventually jumps to a value of 52% accuracy overall, the strongest average from any of our training regimes. The performance eventually falls from 52% over all classes due to the algorithm being constrained to continue selecting batches to train on, moving the weights out of the region of the weight space that achieved 52% accuracy.

For all experiments, the performance is recorded and the best performing model saved as the optimal model. Each method is trained until the onset of overfitting is exhibited. Figures 1 and 2 show performance on the training and validation sets, whereas Figure 3 shows the performance on the validation set. The validation accuracies reported were also found on the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

held-out test set. The optimal network architecture found after cross-validation was a 5 layer deep network with 100 nodes on the hidden layers, all activated by the *'relu'* function. The optimal batch size was 90 for stochastic mini-batch training, the temperature of the output *'softmax'* was 2 and momentum for the stochastic gradient descent was 0.9.

We also experimented using the curricula from [10], [24], [28] and [29] to organise input data. We found that these achieved average accuracies of 40%, 40%, 43% and 42% respectively for curriculum learning, whereas our method achieved 46%. When incorporating the multi-armed bandit, these curricula achieved accuracies of 45%, 46%, 49% and 46%, whereas ours achieved 52%.

To further examine the efficacy of this method, we carry out an experiment using the publicly available MIMIC-III dataset [27, 5]. We see from Figure 4 the stochastic mini-batch training once again providing highly variable performance with a maximum performance of 61%. Figure 5 shows the curriculum regime, once again converging the losses and achieving a better maximum accuracy for all seeds achieving 66%. Finally, Figure 6 shows that our algorithm once again produces the best maximum performance of 69.5% by combining the curriculum regime with the MAB after a brief period of exploration. The curriculum once again smoothes out the losses into a similar minimum in order to provide a stable point from which to launch an exploration of the weight space. The multi-armed bandit then exploits the positioning in the weight space to find a better local minimum. As before the best performing model is saved out before continuing experimentation with the batches. Figures 4 and 5 once again report results on training and validations sets and Figure 6 is displayed only on the validation set. We again find that the reported validation accuracies were also found on the held-out test set. We have therefore shown that this training scheme produces a better performance for two separate classification problems from two separate datasets.

To analyse the performance of our approach across the seven ward-types in the OUH dataset we look at the AUCs of the unique classes after training with the different regimes.

TABLE IV: Maximum performance of various models on ward type prediction for the individual ward types. We test an SVM, feedforward deep neural network trained by stochastic mini-batch training (ff-NN), curriculum learning with a deep neural network (CL) and our proposed method curriculum learning and multi-armed bandit training (CL-MAB). Chance corresponds to an accuracy of 14% and an AUC of 0.5.

Test Data	Model			
	SVM	ff-NN	CL	CL-MAB
Avg. Acc.	0.14	0.39	0.46	**0.52**
Medical AUC	0.50	**0.67**	**0.67**	**0.67**
Cardiac AUC	0.55	**0.78**	**0.78**	**0.78**
Neuro AUC	0.56	0.51	0.56	**0.60**
Trauma AUC	0.66	**0.75**	**0.75**	**0.75**
ICU AUC	0.65	**0.71**	**0.71**	**0.71**
Surgical AUC	0.50	0.59	**0.63**	**0.63**
General/Obs&Gynae AUC	0.54	0.64	0.66	**0.68**

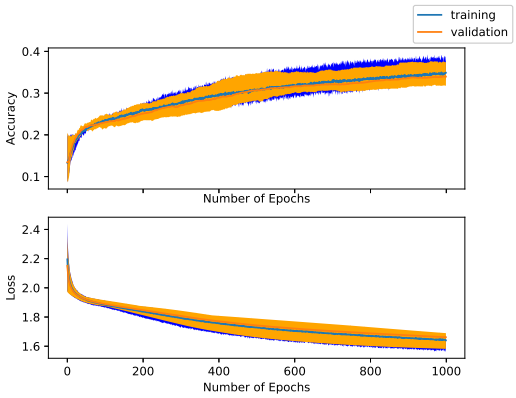


Fig. 1: Batchwise training for five separate seeds. Shaded regions indicate maximum and minimum performance.

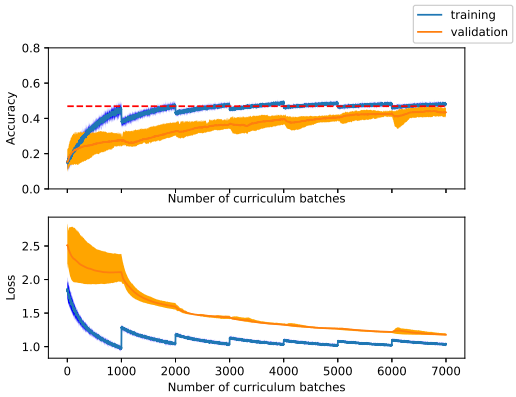


Fig. 2: Mahalanobis curriculum for five separate seeds. Shaded regions indicate maximum and minimum performance. The red line shows the maximum accuracy achieved on the validation set and held-out test set.

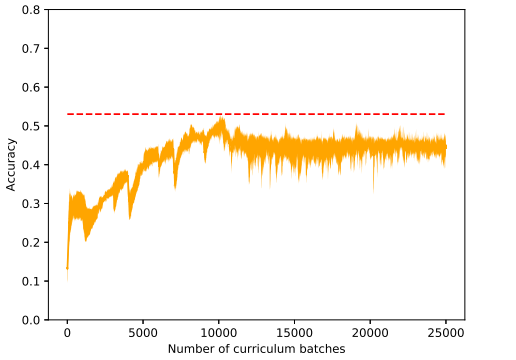


Fig. 3: Curriculum followed by a multi-armed bandit batch selector for five seeds with shaded regions indicating maximum and minimum performance. The validation set is plotted alone for clarity. The red line shows the maximum accuracy achieved on the validation set and held-out test set.

We see that the best performance is achieved by the combination of the curriculum learner and multi-armed ban-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

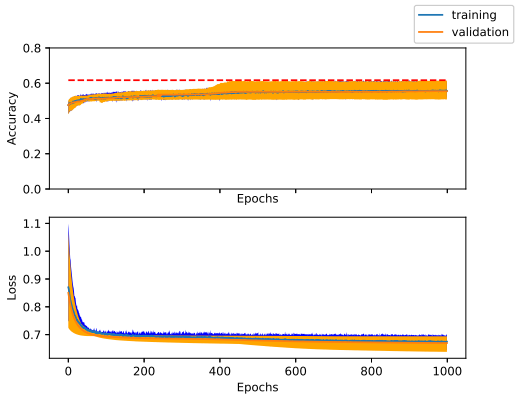


Fig. 4: Batchwise training for five separate seeds on the MIMIC-III dataset. Shaded regions indicate maximum and minimum performance. The red line shows the maximum accuracy achieved on the validation set and held-out test set.

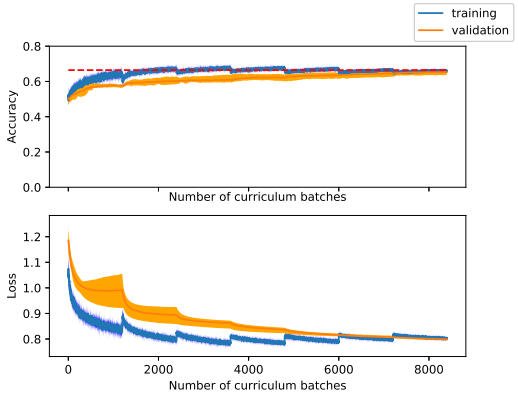


Fig. 5: Mahalanobis curriculum on the MIMIC-III dataset for five separate seeds. Shaded regions indicate maximum and minimum performance. The red line shows the maximum accuracy achieved on the validation set and held-out test set.

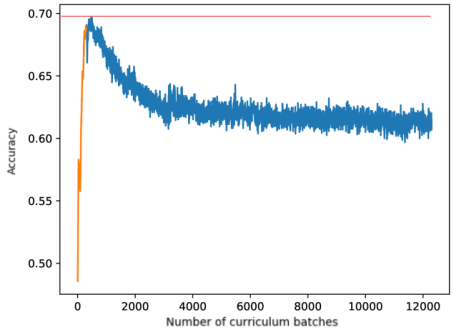


Fig. 6: Curriculum (*orange*) followed by a multi-armed bandit batch selector (*blue*) on the MIMIC-III dataset. The mean performance of the differently seeded models on the validation set is plotted alone for clarity. The red line shows the maximum accuracy achieved on the validation set and held-out test set.

dit, the incorporation of the latter improving the prediction performance on groups 2 and 6 without detriment to the other classes. We further investigate by extracting the latent representation of our test data from the embedded space of the final layer in the network after training. We then apply the t-SNE algorithm [9] to view the clusters that are formed within that space.

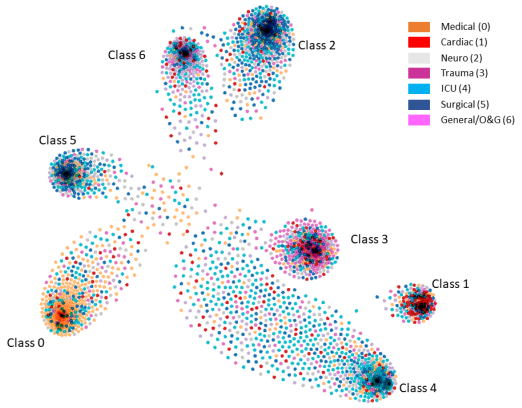


Fig. 7: Visualisation of clustering of the latent representations of the final layer using the t-SNE algorithm.

The result in Figure 7 shows that there are some well defined clusters, coloured by orange, pink, turquoise, red and lilac. However there are two clusters (which correspond to classes 2 and 5) which are not clearly defined by colour and this can be explained as they have low AUC values (see Table IV).

To gain a clearer understanding of why the AUCs for the separate classes are different we use the modified architecture that was described in Section IV-D to interpret feature importance. We retrain the modified architecture to achieve the same accuracy as the previous network while minimising the temperature of the *softmax* from the input layer to achieve as 'peaky' a distribution as possible over the input features. We then extract the trained weights of the inputs, w_{in}. The only features that have weights in the sparse vector (and are therefore considered important for the prediction) are listed in Table V and Figure 8. Table V and Table VII in Appendix A show the binary features which were found to be important for prediction. Figure 8 and Figure 10 in Appendix A show how frequently previous diagnoses appear for patients admitted to a certain ward type. These were compared with the previous diagnoses of the patients admitted to each ward type for the whole dataset and where there was overlap in the diagnoses, these were boxed and labelled as seen in Figure 8b.

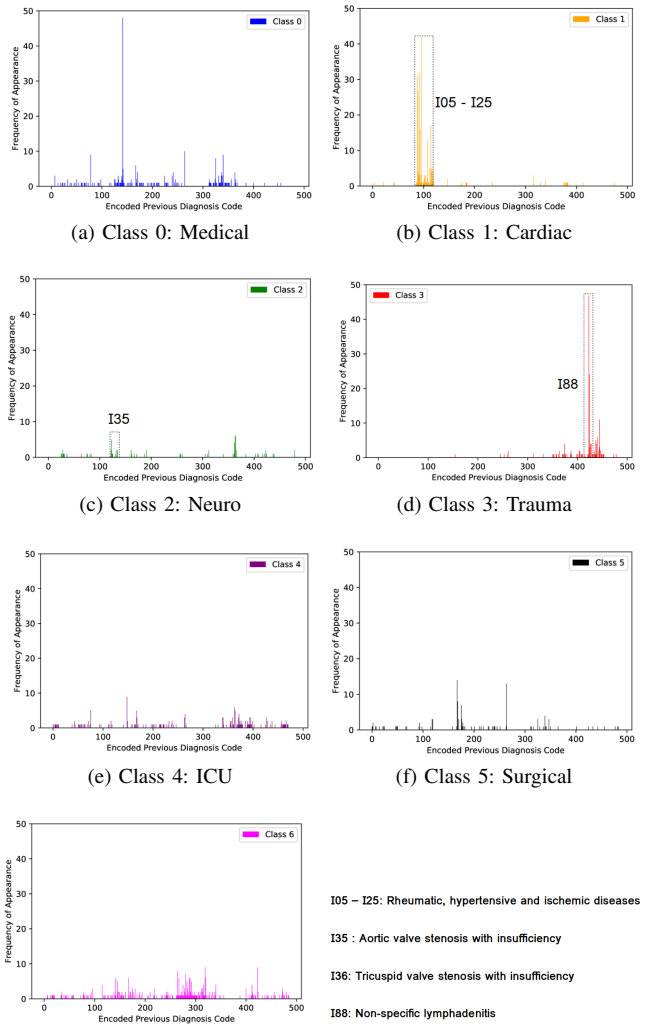
We see from Tables V and VII that the model has learned a distribution based on these 'important' features. These tables explain why the model does not predict accurately for all patients.

- The blood culture test is predominantly carried out for patients who go on to be admitted to classes 0 and 4 which correspond to the '*medical*' wards and the ICUs. This test is used to check for bloodstream infection which can have serious complications and as a result, the model has learned to associate a request for this test with admission under medicine, representing most

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

Feature	Predicted and Actual Class						
	0	1	2	3	4	5	6
Pregnancy test	0.00	0.00	0.01	0.00	0.03	0.00	**0.19**
Blood culture	**0.16**	0.03	0.10	0.02	**0.31**	0.09	0.09
Cardiac enzymes	0.06	**0.58**	0.12	0.09	**0.27**	**0.17**	0.07
Blood cross-match	0.06	0.09	**0.39**	**0.45**	**0.42**	0.00	**0.17**
Frequent Flier	0.04	0.04	0.04	0.02	0.03	**0.22**	0.06

TABLE V: Proportion of patients from each class who had the following tests carried out. All patients in this table were correctly predicted by the model. Values $\geq 15\%$ are highlighted in bold. Classes are: 0 - medical rest, 1 - cardiac, 2 - neurology, 3 - trauma, 4 - ICU, 5 - surgical rest, 6 - general rest.



(a) Class 0: Medical (b) Class 1: Cardiac (c) Class 2: Neuro (d) Class 3: Trauma (e) Class 4: ICU (f) Class 5: Surgical (g) Class 6: General/O&G

Fig. 8: Historical diagnosis code by admission location for correctly predicted patients. Plots are of frequency of appearance by encoded diagnosis code.

patients admitted with an infection, and with the need for intensive care.

- Cardiac enzyme tests are those that are used to indicate a heart attack has occurred or is occurring or if there is blockage in the heart's arteries [2]. It is therefore unsurprising that the model associates this test with class 1, which corresponds to the *'cardiac'* ward types.
- Blood cross-matching (the procedure of searching for appropriate blood to use if a transfusion is required) is a common test asked for from patients who are usually admitted to classes 2, 3 and 4 corresponding to *'neuro'*, *'trauma'* and *'ICU'* ward types respectively. This represents the a subset of patients likely to require surgery during their admission.
- The frequent flier flag is mostly associated with patients admitted to surgical wards (class 5). It is not immediately clear why this is. However, it is hypothesised that this ward function may act as a spare space where beds are available for emptying the ED.
- Pregnancy tests are correlated with the general rest wards (class 6). This likely reflects that admissions under obstetrics and gynaecology fall into this group of patients.

Using the tables and figures we can now see how the predictions are determined.

1) Class 0 (*'Medical'* ward type) are mainly predicted by a blood culture test request and no other tests.
2) Class 1 (*'Cardiac'* ward type) are dominated by having only a cardiac enzyme test requested and no others. Presence of a previous diagnosis of a rheumatic, hypertensive or ischemic disease further increases the likelihood of admission.
3) Class 2 (*'Neuro'* ward type) are predicted by a blood cross-matching request and previous diagnoses, the most prevalent of which correspond to 'aortic valve stenosis with insufficiency'. These are documented in the literature to highly correlate with stroke [6], possibly explaining the reason for these patients' predicted admission to *Neuro*. Upon investigation of the dataset, 86% of the patients who had been previously diagnosed with aortic stenosis would go on to have a subsequent diagnosis associated with cerebral infarction or stroke.
4) Class 3 (*'Trauma'* ward type) is characterised again by a blood cross-match but with different previous diagnoses. In this instance the diagnosis (indicated by the red spikes in Figures 8d and 10d) corresponds to nonspecific lymphadenitis or swelling of the lymph nodes. This is not descriptive enough to gain a physical insight as to why this classification is made. These patients are generally older than the average age of the population of the dataset (65 years old vs. 60 years old generally) and are at a greater risk of previous accidental harm. It is therefore expected that our CL-MAB algorithm has associated a common previous diagnosis code with the greater age of this population and therefore a greater risk of injury. Further investigation would be required to verify that this indeed is the association learned by the algorithm for this patient subset.
5) Class 4 (*'ICU'* ward type) is characterised by a request for blood culture, cardiac enzymes and blood cross-matching. This wide spectrum of tests requested is inidicative of the critical condition the patient is likely

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

to be in upon presentation.

6) Class 5 (*'Surgical'* ward type) is also characterised by a Cardiac Enzyme test requested. It is also not clear if having a 'Frequent Flier' flag causes the prediction.
7) The cause of a prediction of class 6 (*'General / O&G'* ward type) is mainly due to a pregnancy test and this is most likely due to the inclusion of O&G admissions in this ward type.

The overlap in important features for the *'neuro'* and *'trauma'* classes may also explain the difference in AUCs reported in Table IV. It is very possible that many *'neuro'* admissions are predicted to be *'trauma'* due to the similarity in their input importance. This may also be the case for *'surgical'* and *'cardiac'* admissions. To improve our model it will be important to determine if there are further specific features that can be obtained at ED triage time for all classes that may help distinguish these classes.

For comparison we check the distribution of these features for the whole population using the real labels of what ward type each patient was admitted to. The distribution is shown in Table VI.

Feature	Actual Class						
	0	1	2	3	4	5	6
Pregnancy test	0.01	0.00	0.01	0.01	0.02	0.00	0.07
Blood culture	**0.18**	0.05	0.10	0.03	**0.26**	**0.29**	0.12
Cardiac enzymes	0.13	**0.51**	0.09	0.11	**0.25**	**0.24**	0.14
Blood cross-match	0.08	0.11	**0.33**	**0.42**	**0.35**	0.07	**0.16**
Frequent Flier	0.05	0.05	0.05	0.03	0.05	0.05	0.05

TABLE VI: The total population of the dataset is tabulated here using their real ward types as the label. Values $\geq 15\%$ are highlighted in bold.

From Table VI we see that the model has learned the underlying distribution quite accurately. The exceptions are in classes 5 (*'Surgical'*) and 6 (*'General/O&G'*). For Class 6, we see the pregnancy test is not very important for prediction but the blood cross-match is. This motivates the introduction of a gender-specific model. For Class 5 the model has not learned that a blood culture test request as well as a cardiac enzymes test request are most indicative for this class and not the frequent flier flag. This may explain the reason for the poor performance in AUC for class 5. Class 2 (*'Neuro'*) also has a relatively poor performance and based on the distributions in Tables V, VII and VI, it could be due to blood cross-matching tests being important features for classes 3 (*'Trauma'*) and 4 (*'ICU'*) as well. To further improve the performance of the model we will investigate further features that are more specific to the individual ward types, as well as developing separate models for male and female patients. Another limitation of our work is that some patient admissions require specific equipment which can only be found in certain wards [25]. A future model should incorporate this requirement to maximise usefulness of the model to clinicians.

To further examine the usefulness of the model to clinical staff we investigate its performance plotted over time. Figure 9 shows how the model performance varies with time. The red shaded regions indicate the winter flu seasons where the ED gets busiest with admissions. We see that the model does not suffer significant degradation in performance due to winter pressures. In addition in three out of four of the flu seasons the model performs better than the yearly average. We believe this could be due to the grouping of wards into ward functions as opposed to individual wards, which bypasses the problem of patients being admitted to a ward atypical for their condition but still capable of treating the patient. However this may also be due to our preprocessing step of removing patients obviously admitted to an inappropriate ward for their diagnosis. While this filters the obvious cases, it does not remove all such cases from the dataset. We therefore believe that this model could still be useful in helping clinicians during busy periods to request bed space well in advance of the need for it to allow timely admission of patients from the ED and into the hospital ward.

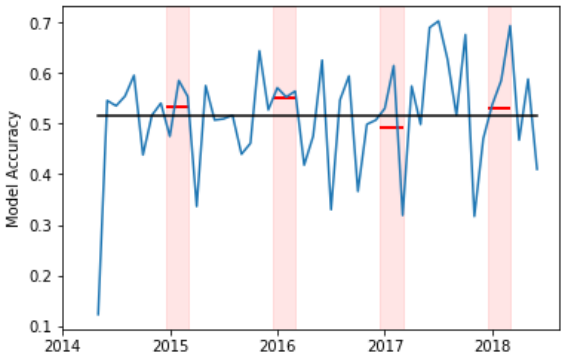


Fig. 9: Performance of the model by month for the 4 years of data included in the dataset. The black solid line is the overall accuracy over the four years. The red shaded area shows the winter flu seasons and the solid red lines show the average performance of the model during those flu seasons.

VII. Conclusion

In this article we have presented a novel method of training and regularising deep learning model with the aim of predicting where a patient presented to the ED will be admitted in an OUH Trust hospital. This prediction will aid in the provision of timely care and treatment for the patient and those still in the ED. Our model achieves AUC values between 0.60 and 0.78 for the individual ward types. Furthermore, our model also provides an explanation as to the cause of the predictions, allowing the user to incorporate more important features for individual ward types in the future. The authors believe this may be useful for ensuring timely admission to hospital and reducing the time to care. This will in turn improve the quality of care for patients still in the ED due to less crowding. This work may also be useful for resource prediction and optimisation in hospitals more generally.

VIII. Future Work

The model presented in this work is first trained using a curriculum and then using the curriculum batches a multi-armed bandit is employed to improve the performance. While

the algorithm described in Algorithm 1 is non-stationary, it is weakly non-stationary relying on the number of pulls of a certain batch to reduce the probability of choosing said batch. As a result, we will improve this by turning this problem into a full reinforcement learning problem. Treating the weights of the network as the state space, we will train a policy to select the best action to take (batch to train on) given the state space. We believe this will be a much more effective method of training due to the information provided to the trainer about the state of the weights of the network.

We would also like to further investigate features that can be obtained from the ED which correlate highly with the individual ward types. In doing so we will be able to reduce the input feature space and advise clinicians in the ED what needs to be measured for this prediction problem. It is hoped that by doing this, we will be able to mitigate the problem of missing features which can commonly happen in models with large input spaces. We will continue investigating methods of identifying when patients were admitted to wards that were not ideal for their treatment. We believe that finding these cases will help to improve the performance of our models due to their reliance on historical data. We will also seek to integrate data on the equipment used during a patient stay to better inform the model of which wards are appropriate for admission.

ACKNOWLEDGMENT

We thank all the people of Oxfordshire who contribute to the Infections in Oxfordshire Research Database. This work uses data provided by patients and collected by the NHS as part of their care and support. Research Database Team (Oxford): R Alstead, C Bunch, DW Crook, J Davies, J Finney, J Gearing (community), H Jones, L O'Connor, TEA Peto (PI), TP Quan, J Robinson (community), B Shine, AS Walker, D Waller, D Wyllie. Patient and Public Panel: G Blower, C Mancey, P McLoughlin, B Nichols.

This work was funded by the EPSRC Doctoral Training Award as part of the UK's high-priority "industrial strategy" funding scheme and the National Institute for Health Research Oxford Biomedical Research Centre. Tingting Zhu was supported by the Royal Academy of Engineering Research Fellowship. David Eyre is a Big Data Institute Robertson Foundation Fellow.

IX. DISCLOSURE

David Eyre has received lecture fees and conference expenses from Gilead.

REFERENCES

[1] Andrew K Diehl, Max D Morris, and Stephen A Mannis. "Use of calendar and weather data to predict walk-in attendance." In: *Southern medical journal* 74.6 (1981), pp. 709–712.

[2] Bruce D McCarthy, John B Wong, and Harry P Selker. "Detecting acute cardiac ischemia in the emergency department". In: *Journal of General Internal Medicine* 5.4 (1990), pp. 365–373.

[3] Eugene Somoza and John R Somoza. "A neural-network approach to predicting admission decisions in a psychiatric emergency room". In: *Medical Decision Making* 13.4 (1993), pp. 273–280.

[4] Roy De Maesschalck, Delphine Jouan-Rimbaud, and Désiré L Massart. "The mahalanobis distance". In: *Chemometrics and intelligent laboratory systems* 50.1 (2000), pp. 1–18.

[5] Ary L Goldberger et al. "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals". In: *Circulation* 101.23 (2000), e215–e220.

[6] Jamary Oliveira-Filho et al. "Stroke as the first manifestation of calcific aortic stenosis". In: *Cerebrovascular Diseases* 10.5 (2000), pp. 413–416.

[7] Peter Auer et al. "The nonstochastic multiarmed bandit problem". In: *SIAM journal on computing* 32.1 (2002), pp. 48–77.

[8] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[9] L.J.P. van der Maaten and G.E. Hinton. "Visualizing High-Dimensional Data Using t-SNE". In: *Journal of Machine Learning Research* 9 (2008), pp. 2579–2605.

[10] Yoshua Bengio et al. "Curriculum learning". In: *Proceedings of the 26th annual international conference on machine learning*. ACM. 2009, pp. 41–48.

[11] Omar Hasan et al. "Hospital readmission in general medicine patients: a prediction model". In: *Journal of general internal medicine* 25.3 (2010), pp. 211–219.

[12] Victor A Convertino et al. "Use of advanced machine-learning techniques for noninvasive monitoring of hemorrhage". In: *Journal of Trauma and Acute Care Surgery* 71.1 (2011), S25–S32.

[13] John Gittins, Kevin Glazebrook, and Richard Weber. *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011.

[14] Mohammed Saeed et al. "Multiparameter Intelligent Monitoring in Intensive Care II (MIMIC-II): a public-access intensive care unit database". In: *Critical care medicine* 39.5 (2011), p. 952.

[15] Eugene L Allgower and Kurt Georg. *Numerical continuation methods: an introduction*. Vol. 13. Springer Science & Business Media, 2012.

[16] Juan F Fernandez, Oriol Sibila, and Marcos I Restrepo. "Predicting ICU admission in community-acquired pneumonia: clinical scores and biomarkers". In: *Expert review of clinical pharmacology* 5.4 (2012), pp. 445–458.

[17] Oğuz Karan et al. "Diagnosing diabetes using neural networks on small mobile devices". In: *Expert Systems with Applications* 39.1 (2012), pp. 54–60.

[18] José Labarère et al. "Validation of a clinical prediction model for early admission to the intensive care unit of patients with pneumonia". In: *Academic Emergency Medicine* 19.9 (2012), pp. 993–1003.

[19] Robert Francis. *Report of the Mid Staffordshire NHS Foundation Trust public inquiry: executive summary*. Vol. 947. The Stationery Office, 2013.

[20] P Gueth et al. "Machine learning-based patient specific prompt-gamma dose monitoring in proton therapy". In: *Physics in Medicine & Biology* 58.13 (2013), p. 4563.

[21] Znaonui Liang et al. "Deep learning for healthcare decision making with EMRs". In: *2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE. 2014, pp. 556–559.

[22] Kagobora Pascasie and Ntombifikile Gloria Mtshali. "A descriptive analysis of emergency department overcrowding in a selected hospital in Kigali, Rwanda". In: *African Journal of Emergency Medicine* 4.4 (2014), pp. 178–183.

[23] Haijun Zhai et al. "Developing and evaluating a machine learning based algorithm to predict the need of pediatric intensive care unit transfer for newly hospitalized children". In: *Resuscitation* 85.8 (2014), pp. 1065–1071.

[24] Anastasia Pentina, Viktoriia Sharmanska, and Christoph H Lampert. "Curriculum learning of multiple tasks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 5492–5500.

[25] Clair Sullivan et al. "National Emergency Access Targets metrics of the emergency department–inpatient interface: measures of patient flow and mortality for emergency admissions to hospital". In: *Australian Health Review* 39.5 (2015), pp. 533–538.

[26] Paul Richard Edwin Jarvis. "Improving emergency department patient flow". In: *Clinical and experimental emergency medicine* 3.2 (2016), p. 63.

[27] Alistair EW Johnson et al. "MIMIC-III, a freely accessible critical care database". In: *Scientific data* 3 (2016), p. 160035.

[28] Srikar Appalaraju and Vineet Chaoji. "Image similarity using deep CNN and curriculum learning". In: *arXiv preprint arXiv:1709.08761* (2017).

[29] Alex Graves et al. "Automated curriculum learning for neural networks". In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 1311–1320.

[30] Fenglong Ma et al. "Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks". In: *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*. ACM. 2017, pp. 1903–1911.

[31] Tambet Matiisen et al. "Teacher-student curriculum learning". In: *arXiv preprint arXiv:1707.00183* (2017).

[32] Riccardo Miotto et al. "Deep learning for healthcare: review, opportunities and challenges". In: *Briefings in Bioinformatics* 19.6 (May 2017), pp. 1236–1246.

[33] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. "Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models". In: *arXiv preprint arXiv:1708.08296* (2017).

[34] Andrew Staib et al. "Uniting emergency and inpatient clinicians across the ED–inpatient interface: The last frontier?" In: *Emergency Medicine Australasia* 29.6 (2017), pp. 740–745.

[35] Hongteng Xu et al. "Patient flow prediction via discriminative learning of mutually-correcting processes". In: *IEEE transactions on Knowledge and Data Engineering* 29.1 (2017), pp. 157–171.

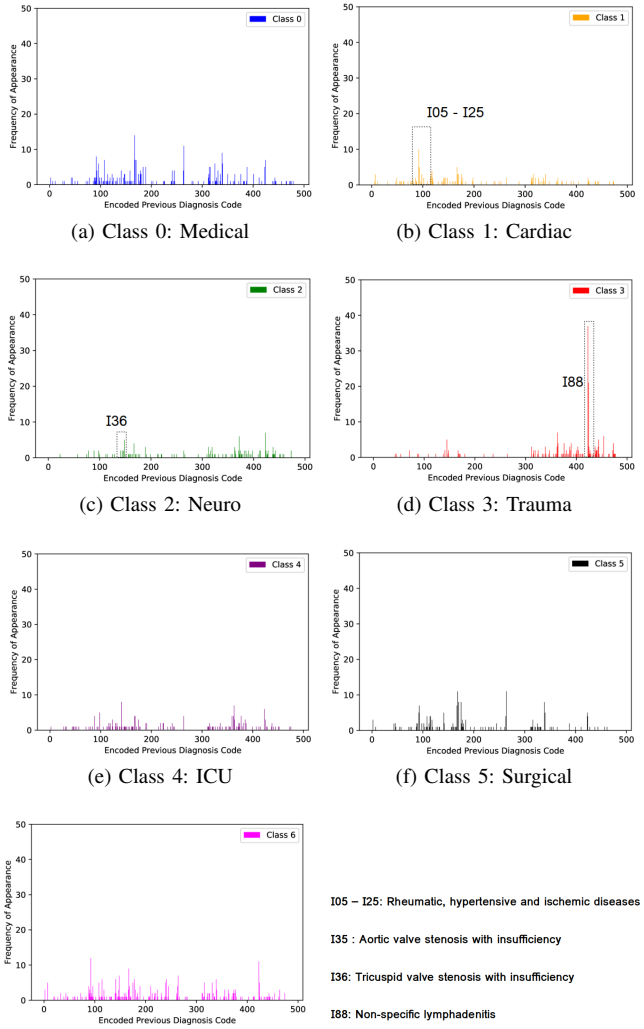
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2020.2990309, IEEE Journal of Biomedical and Health Informatics

APPENDIX A
INCORRECT PREDICTIONS DISTRIBUTION

For comparison, we show the distribution of the features of patients who were predicted to be one of these classes but the classification was incorrect. These results are shown in Table VII and Figure 10.

Feature	Predicted and not Actual Class						
	0	1	2	3	4	5	6
Pregnancy test	0.00	0.00	0.00	0.00	0.02	0.00	0.02
Blood culture	**0.38**	0.03	0.09	0.03	**0.33**	0.07	**0.16**
Cardiac enzymes	0.10	**0.35**	0.05	0.04	**0.21**	**0.15**	0.05
Blood cross-match	0.07	0.08	**0.23**	**0.39**	**0.70**	0.00	0.10
Frequent Flier	0.06	0.03	0.02	0.04	0.04	0.04	0.08

TABLE VII: All patients in this table were incorrectly predicted by the model to belong to these classes. Values $\geq 15\%$ are highlighted in bold.



(a) Class 0: Medical
(b) Class 1: Cardiac
(c) Class 2: Neuro
(d) Class 3: Trauma
(e) Class 4: ICU
(f) Class 5: Surgical
(g) Class 6: General/O&G

Fig. 10: Historical diagnosis code by admission location for incorrectly predicted patients. Plots are of frequency of appearance by encoded diagnosis code.