Article

A Fairness of Data Combination in Wireless Packet Scheduling

Sovit Bhandari ¹ , Navin Ranjan ¹ , Yeong-Chan Kim ^{1,2}, Pervez Khan ² and Hoon Kim ^{1,2,*}

¹ IoT and Big-Data Research Center, Incheon National University, Yeonsu-gu, Incheon 22012, Korea; sovit198@gmail.com (S.B.); ranjannavin07@gmail.com (N.R.); kyc0288@inu.ac.kr (Y.-C.K.)

² Department of Electronics Engineering, Incheon National University, Yeonsu-gu, Incheon 22012, Korea; pervazkanju@hotmail.com

* Correspondence: hoon@inu.ac.kr

Abstract: With the proliferation of artificial intelligence (AI) technology, the function of AI in a sixth generation (6G) environment is likely to come into play on a large scale. Moreover, in recent years, with the rapid advancement in AI technology, the ethical issues of AI have become a hot topic. In this paper, the ethical concern of AI in wireless networks is studied from the perspective of fairness in data. To make the dataset fairer, novel dataset categorization and dataset combination schemes are proposed. For the dataset categorization scheme, a deep-learning-based dataset categorization (DLDC) model is proposed. Based on the results of the DLDC model, the input dataset is categorized based on the group index. The datasets based on the group index are combined using various combination schemes. Through simulations, the results of each dataset combination method and their performance are compared, and the advantages and disadvantages of fairness and performance according to the dataset configuration are analyzed.

Keywords: artificial intelligence; ethical; fairness; packet scheduling; wireless networks; convolutional neural network

Citation: Bhandari, S.; Ranjan, N.; Kim, Y.-C.; Khan, P.; Kim, H. A Fairness of Data Combination in Wireless Packet Scheduling. *Sensors* **2022**, *22*, 1658. <https://doi.org/10.3390/s22041658>

Academic Editor: Qammer Hussain Abbasi

Received: 25 January 2022

Accepted: 18 February 2022

Published: 20 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Motivation

With the rapid growth and proliferation of artificial intelligence (AI) technologies and applications, the question of ethical AI has arisen. In recent years, a large number of works have been intensively studied on the principles and guidelines of AI ethics. AI fairness is one of the key elements of the principles or guidelines, and data fairness in AI learning is considered as one of the main concerns [1]. The work in [2] offers a new approach: a set of principled solutions based on the emerging and exciting science of socially aware algorithm design. The authors explain how we can better embed human principles into machine code—without halting the progress of data-driven scientific exploration. The work in [3] examined several real-world applications that exhibited biases in different ways and listed several sources of biases that can affect AI applications. The authors also created a taxonomy for fairness definitions that machine learning researchers have defined to avoid the existing biases in AI systems.

Since the sixth generation (6G) wireless network system is likely to integrate AI technologies into its system [4], packet scheduling is one of the most important areas of the wireless system where the ethical aspects of AI should be considered, as it is responsible for selecting user equipment (UEs) to transmit data packets. The goal of packet scheduling in a wireless network system is to ensure efficient and fair resource utilization for all UEs [5–7]. Some of the popular traditional approaches to packet scheduling in wireless networks include: round-robin (RR), maximum system throughput (MST), and proportional fair scheduling (PFS) [8–11].

In the RR-based scheme, UEs are selected in a round-robin fashion so that all UEs have the equal probability of being selected in the time domain despite radio channel

status. This type of scheme cannot perform adaptive adjustment according to change in channel condition. To address the drawback of RR, MST was proposed. In an MST-based scheme, the UEs with the best channel are selected to maximize the total system throughput in each scheduling epoch. This strategy sacrifices the fairness among UEs because those UEs with poor channel conditions have high likelihood of not being selected. To address the drawbacks of both RR- and MST-based schemes, PFS was proposed by F. Kelly [10]. PFS, a widely used scheduling scheme, provides a tradeoff between overall system throughput and fairness among UEs selection. The works described in [8–11] are iterative-based schemes, which are more difficult to implement in a real-time network with a large number of system dynamics, since they require a comparatively high computational effort [12].

In recent years, there has been major breakthroughs in the field of wireless communications using AI technology [13–15]. Some of the works which have considered AI to solve packet scheduling problems in wireless network systems are listed in [16–19]. In [16], a support vector machine (SVM) model-based packet scheduling in wireless communication networks has been proposed. The SVM model is fed on the data produced by the PFS scheme considering only two UE scenario, which is not practical for real life implications. In [17], a supervised deep neural network (DNN)-based method is proposed to schedule channel in interference-limited wireless networks. The DNN is fed on the data produced by the iterative-based sequential convex approximation (SCA) algorithm. In [18], the joint subcarrier and power allocation problem is studied, where a deep Q-network is employed for scheduling decision making and a DNN is trained for power allocation using supervised learning. In [19], deep reinforcement learning (DRL)-based double-deep Q network (DDQN) framework is proposed to make scheduling decisions in the edge computing environment. However, the works mentioned in [16–19] more focused on increasing overall system throughput and did not guarantee the selection of UEs having low channel strength.

Motivated by the growing interest in ethical AI [1–3], in order to guarantee fairness in the UEs selection procedure for packet scheduling while maximally maintaining existing wireless system performance, a novel dataset categorization and dataset combination schemes are proposed in this paper.

1.2. Related Work

In this sub-section, we present some of the related work that deals with maintaining as well as investigating the fairness of data in the AI domain and are listed in [20–27].

In [20], the fairness of the recidivism prediction instrument is discussed and shows the differential impact of using a recidivism prediction instrument that is not known to have prediction errors. In [21], a regression-based approach is presented to remove the implicit bias present in the correctional offender management profiling for alternative sanctions (COMPAS) dataset. The COMPAS dataset includes suspect records from Broward County with information on their jail and prison time, demographics, criminal history, and COMPAS risk scores [28]. Similarly, in [22], an unbiased regression-based approach is presented to predict the actual assessed targets by ensuring impartiality in terms of gender and race.

In [23], a data profiling framework based on deep learning and statistical model algorithms is proposed to improve the quality of data by finding outliers. This framework was validated using a salary dataset published by the state of Arkansas. In [24], a counterfactual fairness framework that uses tools of causal inference to ensure group fairness (treating different groups equally) is presented. In [25], a criterion for discrimination against a specified sensitive attribute in supervised learning is proposed to predict some target based on available features. The framework is tested on the problem of fairly predicting the acceptance of law students. In addition, in [26,27], an AI-based real-time toolkit for fairness assurance is presented.

In [26], the Aequitas toolkit is presented, which allows users to test models with respect to various bias and fairness metrics for different population subgroups. In [27],

the AI Fairness 360 (AIF360) toolkit developed by IBM is presented. This toolkit helps transfer fairness research algorithms into an industrial setting and provides a benchmark for fairness algorithm evaluation. Moreover, it also provides a platform for fairness researchers to share their ideas.

The work presented in [20–27] addressed fairness with respect to various demographic groups, which cannot be used to ensure fairness in wireless data because they do not contain demographic information. Inspired by the group fairness shown in [20–27], this work proposes a wireless dataset $\mathcal{D}$ categorization and combination scheme to ensure fairness in selecting UEs for packet transmission at any scheduling epochs.

1.3. Contribution and Organization

In this paper, a wireless dataset $\mathcal{D}$ categorization scheme and various combination schemes are proposed to give user candidates with poor channel status the same opportunity as user candidates with good/medium channel status. The contribution of this paper can be summarized as follows:

1. In the first part, a deep learning (DL)-based data categorization (DLDC) scheme is proposed to categorize the wireless dataset $\mathcal{D}$, which contains the timestamp, information about the connecting UEs, information about the base transceiver station (BTS), and information about the channel capacity between the connecting UEs and the BTS, into three different groups. The categorization is performed on the basis of the channel capacity between the connecting BTS and UEs.
2. In the second part, different data combination schemes are proposed to combine the group-index-based categorized wireless dataset $\mathcal{D}$ using a DLDC scheme such that the UEs having poor channel capacity are also included for the data transmission process. In the data combination schemes, different methods are proposed to show how the UEs selection process is affected when there is an imbalance in the dataset so that we can recognize the importance of the ethical dataset.

The remainder of this paper is organized as follows: Section 2 describes the schemes for categorizing and combining wireless dataset $\mathcal{D}$. Section 3 introduces the DLDC scheme. Section 4 evaluates the performance of the proposed DLDC model, and based on which the results of the data combination schemes are shown. Section 5 discusses the overall results and limitations of this work. Finally, in Section 6, conclusions are drawn.

2. Dataset Categorization and Dataset Combination Schemes

In this section, wireless system dataset categorization and combination schemes are proposed to maintain fairness in UE selection while maintaining system throughput for data transmission in the next scheduling epoch. To propose a fairness in UE selection procedure for packet transmission, a wireless system environment consisting of M -antenna BTS is considered as shown in Figure 1 to create the wireless system dataset. Since fairness can be modeled in the resource-deficit scenario, in the figure, it is shown that the single BTS can serve N_s UEs out of N_a UEs at any time instance t .

2.1. Dataset Categorization Scheme

In this part, the wireless dataset $\mathcal{D}$ obtained from the system environment is divided into three different groups, namely, Group 1 (G_1), Group 2 (G_2), and Group 3 (G_3), as shown in Figure 1. Since the UEs are constantly moving in the wireless system environment, the UEs that belong to one group in one time period may belong to another group in another time period. Therefore, AI is required to determine the group index G_i ($i \in \{1, 2, 3\}$) of the connecting UEs instantaneously. Since the 6G wireless system is likely to embed AI functions into its system, in this paper, the need for AI is satisfied by the DLDC model, which is a DL-based 2D convolutional neural network (CNN) model. The categorization of the wireless dataset $\mathcal{D}$ on the basis of group index G_i can be performed by employing it to the trained DLDC model, as shown in Figure 2.

Figure 1. A wireless system environment containing M -antenna BTS serving N_s UEs out of N_a UEs.

Figure 2. A wireless system dataset categorization scheme based on DLDC.

In Figure 2, the category G_1 dataset contains channel information of the connecting UEs that exhibit good channel strength while connecting to the M -antenna BTS at any time instance t . Similarly, the category G_2 dataset contains channel information of the connecting UEs that have medium channel strength while connecting to the M -antenna BTS at any time instance t . Likewise, the category G_3 dataset contains channel link information of the connecting UEs that have poor channel strength while connecting to the M -antenna BTS at any time instance t .

The detailed explanation of the DLDC scheme to categorize the dataset into multiple groups is discussed in Section 3.

After the categorization of the dataset containing the different channel strength information between the connecting UEs and M -antenna BTS into three groups, the multiple dataset combination schemes are provided in the next sub-section to maintain the UEs selection fairness while maintaining overall system throughput.

2.2. Dataset Combination Schemes

In this sub-section, the dataset obtained from the dataset categorization scheme, i.e., G_1 , G_2 , and G_3 , is merged using different dataset combination schemes (see Figure 3) so that the fairness of UE selection is maintained.

Figure 3. A general wireless system dataset combination scheme.

In Figure 3, only a selected portion of the dataset from each group is combined to meet the requirements of the wireless system environment and to make the data more ethical. As shown in the figure, part P_1 of the G_1 dataset, part P_2 of the G_2 dataset, and part P_3 of the G_3 dataset are randomly selected before they are combined. This ensures that the UEs from each group are given priority for data transmission despite the channel strength, provided it meets the minimum requirements for data transmission.

Based on the values of the selection portion P_1 , P_2 , and P_3 , the dataset combination schemes S_k ($k \in \{1, 2, 3\}$) are classified into three different schemes, namely, the random dataset combination scheme (RDCS) or Scheme 1 (S_1), the equal dataset combination scheme (EDCS) or Scheme 2 (S_2), and the weighted dataset combination scheme (WDCS) or Scheme 3 (S_3), as shown in Figure 3.

Based on the values of the selection portion P_1 , P_2 , and P_3 , the data combination scheme can be further divided into the three different schemes, namely, random dataset combination scheme (RDCS), equal dataset combination scheme (EDCS), and weighted dataset combination scheme (WDCS), as shown in Figure 4.

Figure 4. Types of wireless dataset combination schemes.

The different kinds of dataset combination schemes are briefly described in the sub-sections of this part.

2.2.1. Random Dataset Combination Scheme

In this scheme, the dataset containing the radio link information of the connecting UEs and the BTS is randomly combined without any group information index. The fairness in the selection of UEs is not guaranteed in this scheme.

2.2.2. Equal Dataset Combination Scheme

In this scheme, an equal proportion of the dataset from each group is used before combination, i.e., the ratio of $P_1: P_2: P_3$ is set to 1:1:1. In this scheme, fairness is ensured in the selection of UEs, since the UEs with the medium and poor channel strength are also preferred in the data transmission process.

2.2.3. Weighted Dataset Combination Scheme

In this scheme, an unequal proportion of the dataset from each group is used before the combination, e.g., the ratio of $P_1: P_2: P_3$ is set to 1:2:3. If the ratio of $P_1: P_2: P_3$ is set to 1:2:3, it means that the candidate of Group 3 related to weak channel strength will receive higher priority in data transmission compared to the other groups.

3. Deep-Learning-Based Dataset Categorization Scheme

In this section, the DLDC scheme, a 2D CNN-based DL model, is proposed to categorize datasets into multiple groups. The dataset with input features such as the timeslot, information about connecting UEs, the connecting BTS, and the channel capacity between connecting UEs, and BTS is fed into the DLDC scheme to categorize the dataset into multiple groups. Before applying DLDC to categorize the dataset, the dataset is generated for the DLDC scheme. The dataset generation procedure is described in Section 3.1.

3.1. Dataset Generation

In this sub-section, we create a dataset $\mathcal{D}$ that will be used for the DLDC model. Based on the system model shown in Figure 1, a dataset is created. It is assumed that there are 12 UEs that attempt to connect to the M -antennas BTS at each time t . The link capacity between the UEs and the BTS is determined by the Shannon capacity limit, i.e., $B \times \log_2(1 + SIR)$, where B is the channel bandwidth and SIR is the signal-to-interference ratio. It is assumed that the channel distribution of the air interface link is a Rayleigh distribution. Since large dataset is required for training the supervised DL-based model [29], the 0.1 million (M) timeslot is considered to generate moderately large dataset [30] for the 12 UEs scenario to avoid the computational effort. In [31,32], it is reported that the maximum downlink spectral efficiency for 5G is 30 bps/Hz, and for 6G, it will be 100 bps/Hz. Based on [31,32], we assumed the average SIR of our wireless system environment to be 33 dB when creating the 0.1 M 2D dataset. Since we considered the supervised 2D

CNN-based learning method, the label in the dataset is created based on the value of SIR . If the SIR of the channel is greater than 33 dB, it is considered to be a good channel, so the UEs under this group are labelled as G_1 . If the SIR of the channel is less than 27 dB, it is considered to be a bad channel, and the UEs under this group are called G_3 . The channel that lies between the good and the bad channel is called the middle channel and is represented by G_2 . The G_1 , G_2 , and G_3 labels are identified by the three-bit combination, i.e., [1,0,0], [0,1,0], and [0,0,1], respectively. The three-bit combination is obtained by using one-hot encoding method [33]. The simulation parameters used to create the dataset are given in Table 1.

Table 1. Simulation parameters for generating dataset.

Simulation Parameters	Value
UEs Requesting for Connection (N_a)	12
Good Channel Condition	$SIR \text{ (dB)} \geq 33$
Medium Channel Condition	$27 \leq SIR \text{ (dB)} < 33$
Bad Channel Condition	$SIR \text{ (dB)} < 27$
Average SIR	33 dB
Channel Variation Distribution	Rayleigh distribution
Channel Capacity	$B \times \log_2(1 + SIR)$
Total Timeslot	0.1 M

3.2. Model Implementation

In DLDC scheme, we have used 2D-based CNN model for feature extraction from input dataset $\mathcal{D}$. The main goal of this model is to categorize input dataset into G_1 , G_2 , and G_3 groups, respectively, based on the input features such as timeslot, user information, and channel capacity. Since we have only assumed single BTS in our system, the information regarding BTS is avoided from the dataset while training the model.

In the DLDC scheme, we have used a 2D-based CNN model for feature extraction from the input dataset $\mathcal{D}$. The main objective of this model is to categorize the input dataset $\mathcal{D}$ into G_1 , G_2 , and G_3 groups, based on the input features such as the timeslot, user information, and channel capacity. Since we have only assumed a single BTS in our system, the information regarding BTS is avoided from the dataset while training the model.

The input data per timeslot are converted to the 2D format having the dimensions of $12 \times 3 \times 1$ (height $\times$ width $\times$ channel) before employing it to the DLDC model. The height in the data depends upon the number of UEs, and the width represents the number of the input features (i.e., 3). The 70% of $\mathcal{D}$, i.e., 0.07 M data of size 12×3 , is used for training the model, while 20% of $\mathcal{D}$ (0.02 M) is used for validation. The remaining 10% of $\mathcal{D}$ (0.01 M) is used for testing the trained model. The 2D CNN model used in this paper is shown in Figure 5.

Figure 5. DLDC model for categorization of dataset into different groups.

In Figure 5, the architecture of the DLDC model is formed by stacking four convolutional layers. First, the 2D input of size 12×3 (row number $\times$ column number) is used in the first 2D convolutional layer of the DLDC model shown to produce an output of size $12 \times 3 \times 12$. In the first convolutional layer, the input is scaled up using 12 filters of size 2×2 with a stride of 1×1 . In the convolved output of the first layer, batch normalization (BN) is used, and then it is passed to the rectified linear unit (ReLU). In our DLDC model, BN is used to stabilize the learning process and reduce the number of epochs required to train the neural networks [34], and the ReLU activation function is used to increase the nonlinearity of the input data and solve the vanishing gradient problem [35]. The output of the ReLU activation function of the first convolution layer is then used in the second convolution layer with filter size eight for reduction. In this layer, the size is reduced to $12 \times 3 \times 8$. In the convolved output of the second convolution layer, BN is applied and then fed to the ReLU activation. The output of the ReLU activation function of the second convolution layer is fed to the third convolution layer. Since only four filters are used in the third convolution layer, the output of the third convolution layer is reduced to $12 \times 3 \times 4$. BN is applied to the output of the third convolution layer, and then passed to the ReLU activation function. The output of the ReLU activation function in the third layer is used in the last convolution layer. In the last layer, the input is convolved to produce an output of size $12 \times 3 \times 1$ by using only one filter in this layer. Since our problem is a multi-label classification problem, the binary cross-entropy (BCE) loss function is used for training the model along with sigmoid activation function in the last convolution layer [36]. The output of our last layer is sent to the sigmoid activation function. The sigmoid activation function classifies our input data into multiple labels, i.e., G_1 , G_2 , and G_3 , respectively.

In our DLDC scheme, we trained 0.07 M data of size 12×3 to achieve the multi-label classification with better accuracy. Moreover, the Adam optimizer is used to update the weight and learning rate values as it is straightforward to implement, computationally efficient, and has little memory requirements [37].

The detailed structure of our DLDC model is shown in Table 2.

Table 2. Detailed structure of DLDC with four convolutional layers.

Layer Name	Input Size	Output Size	Filter Size
Conv2D_1	$12 \times 3, 1$	$12 \times 3, 12$	$2 \times 2, 12$
Conv2D_2	$12 \times 3, 12$	$12 \times 3, 8$	$2 \times 2, 8$
Conv2D_3	$12 \times 3, 8$	$12 \times 3, 4$	$2 \times 2, 4$
Conv2D_4	$12 \times 3, 4$	$12 \times 3, 1$	$2 \times 2, 1$

4. Performance Evaluation

In this section, the simulation results are shown to evaluate the performance of our proposed DLDC scheme, as well as dataset combination methods such as RDCS, EDCS, and WDCS. The key performance index (KPI) of the proposed DLDC scheme is presented in terms of complexity analysis, training accuracy, test accuracy, BCE loss in the training phase, and BCE loss in the test phase. Based on the results of the DLDC scheme, RDCS, EDCS, and WDCS are applied. The KPI of RDCS, EDCS, and WDCS is shown in terms of the user selection count, fairness in user selection, and average user throughput.

To train the DLDC model, we used the Keras library on top of the TensorFlow framework in Python 3.7 as a programming platform. The training process for our datasets is performed by using a computation server (MiruWare, Seoul, Korea). The specification of the computational server includes; one Intel Core i7 CPU, four Intel Xeon E7-1680 processors, and 128 GB random access memory. The results are obtained by using a computer with 16 GB random access memory and an Intel Core i7-8700 processor. The results are used to apply different data combination schemes.

4.1. DLDC Model KPI

In this part, the DLDC model in terms of training accuracy, test accuracy, BCE loss in the training phase, and BCE loss in the test phase is presented. The proposed 2D CNN-based model is trained using 0.07 M 2D data. Likewise, the validation of the trained model is performed on the 0.02 M 2D data. Finally, the trained model is tested on 0.01 M 2D data. The simulation parameter used while training the model is summarized in Table 3.

Table 3. Simulation parameters for training DLDC model.

Simulation Parameters	Values
Training Size	(70,000, 12, 3, 1)
Validation Size	(20,000, 12, 3, 1)
Testing Size	(10,000, 12, 3, 1)
Number of 2D CNN Layer	4
Number of Features	3
Number of Label	3
Batch Size	32
Learning Rate	0.0001
Epoch	1–10

The training accuracy and test accuracy of the trained DLDC scheme are shown in Figure 6.

Figure 6. DLDC model training as well as testing accuracy for 10 training epochs.

Figure 6 shows that the training accuracy of the DLDC model is 97.5%, and the test accuracy is 96.7% when the model is trained for 10 training epochs. From Figure 6, we can infer that the test accuracy of the DLDC model approaches the training accuracy when the model is trained for up to 10 training epochs and settles at 96 to 97 (%).

The training BCE loss and test BCE loss of the trained DLDC scheme are shown in Figure 7.

Figure 7 shows that the training BCE loss of the DLDC model is 0.0923, and the test BCE loss is 0.1355 when the model is trained for 10 training epochs. From Figure 7, we can infer that the test BCE loss of the DLDC model approaches the training BCE loss when the model is trained for up to 10 training epochs and settles at 0.1 to 0.15.

From Figures 6 and 7, we can conclude that the DLDC model is well trained and can be further used for the categorization of dataset $\mathcal{D}$ in terms of group index G_i .

Figure 7. DLDC model training as well as testing binary cross entropy loss for 10 training epochs.

4.2. Data Combination Schemes KPI

In this part, the performance of the RDCS, EDCS, and WDCS is shown in terms of complexity analysis, user selection count, user selection fairness, and average throughput based on the group-index based dataset, i.e., G_1 , G_2 , and G_3 , provided by the DLDC scheme.

Fairness in the selection of UEs in a wireless system environment for packet scheduling comes into play when the number of UEs attempting to connect to the BTS is greater than the BTS can actually serve in any scheduling epoch. Thus, to create a scenario that addresses the problem of fairness in the selection of UEs in the wireless system environment, the simulations assume that the BTS with M antennas can serve only 6 UEs in any scheduling epoch and that the number of UEs attempting to connect to the BTS at each time t is twice the number of UEs that the BTS can actually serve, i.e., 12. Since the size of the 2D training dataset for the DLDC model depends on the number of UEs attempting to connect to the BTS, a smaller number of UEs is assumed to avoid computational overhead in creating the wireless dataset $\mathcal{D}$ as well as training the DLDC model.

The channel bandwidth of the subcarriers is assumed to be 20 MHz. The minimum required data volume is assumed to be the same for all users in a given timeslot. The performances of the proposed data selection schemes are provided in terms of the 100 scheduling epochs.

4.2.1. Complexity Analysis

In this part, we compare the complexity of RDCS, EDCS, and WDCS. Computational complexity is defined as the number of complex operations. The complex operations considered in the complexity analysis are complex addition, complex multiplication, and complex division [38]. The complexity of our proposed methods for combining datasets is shown in Table 4.

Table 4. Complexity comparison of dataset combination schemes.

Algorithm	Complexity
RDCS	$O(TN_a)$
EDCS	$O(TN_a)$
WDCS	$O(TN_a)$

T is the total timeslot.

From Table 4, we can see that the time complexity for RDCS, EDCS, and WDCS is same, i.e., $O(TN_a)$.

4.2.2. User Selection Count

For RDCS, the role of group index G_i does not come into play, so the data extraction proportion, i.e., P_1 , P_2 , and P_3 . Under RDCS, the UEs are assigned randomly. Based on the given simulation parameters, the total UEs that can be selected under RDCS for 100 is shown in Figure 8.

Figure 8. Total UEs selection count under RDCS for 100 timeslots.

In Figure 8, the total number of UEs selected at each timeslot is six. Since the BTS can serve six UEs at each timeslot, we can say that under RDCS, the UEs slot is fully utilized.

For EDCS, the group index is taken into account. For each group, a maximum of two users are allowed to connect. This is because our assumed system can serve six UEs simultaneously. The total number of users that can be served by this scheme for each scheduling epoch is shown in Figure 9. In Figure 9, the number of UEs selected for packet transmission is either five or six in most timeslots, four in a few timeslots, and three or two in rare cases.

In WDCS, the group index is also considered. However, in this scheme, an unequal proportion of UEs from each group is selected before combination. In our case, we set the UEs selection limit for G_1 to 1, for G_2 to 2, and for G_3 to 3, since the maximum connection limit in each time instance is set to 6. The candidate of G_3 receives a higher priority than the other groups. The total number of users that can be served by this scheme for each scheduling epoch is shown in Figure 10.

In Figure 10, the number of UEs selected for packet transmission is three to six in most timeslots, and rarely, the number of UEs selected is one or two.

In EDCS and WDCS, the UEs slots in most time instances are not as heavily occupied as in RDCS because there are fewer or no UEs in groups G_2 and G_3 in those time instances despite higher preference. By comparing Figures 8–10, we can say that the RDCS is better in terms of UEs selection count.

Figure 9. Total UEs selection count under EDCS for 100 timeslots.

Figure 10. Total UEs selection count under WDCS for 100 timeslots.

4.2.3. User Selection Fairness

In this part, the G_i based user selection fairness factor (f) for each dataset combination schemes S_k is calculated as following:

$$f_{G_i}^{S_k} = \frac{\text{Portion of UEs selected for each group } G_i \text{ under scheme } S_k}{\text{Probability of UEs selection for each group } G_i \text{ under scheme } S_k} \quad (1)$$

where the portion of UEs selected for each group G_i under scheme S_k refers to the ratio of the total selected UEs for the data transfer process belonging to the group G_i under scheme S_k to the total available UEs belonging to the same group under the same scheme in a time period T . The fairness factor (f) for user selection calculated for different dataset

combinations such as RDCS, EDCS, and WDCS based on the group index G_i is shown in Figure 11.

Figure 11. The fairness factor (f) for user selection for each group index G_i under dataset combination schemes such as RDCS, EDCS, and WDC in 100 timeslots.

In Figure 11, we can see that for RDCS, the values of f for G_1 , G_2 , and G_3 are 1.08, 0.89, and 0.87, respectively. In RDCS, the probability of selecting UEs is set to 0.5 because only 6 UEs out of 12 UEs are allowed for the data transmission process in each scheduling epoch. Since the value of f for G_1 is greater than one, while the value of f for G_2 and G_3 is less than one, RDCS is more suitable for selecting G_1 UEs.

Likewise, we can see that for EDCS, the values of f for G_1 , G_2 , and G_3 are 0.88, 1.59, and 2.51, respectively. In EDCS, the probability of selecting UEs belonging to G_1 , G_2 , and G_3 is set to 0.33 for all groups because the same proportion of UEs is assigned to each group. Since the value of f for G_2 and G_3 is greater than one while the value of f for G_1 is less than one, EDCS is more suitable for the selection of G_2 and G_3 UEs. In this scheme the value of f for G_1 is less than one because of the higher counts of UEs belonging to G_1 and comparatively lesser slot allocation for G_1 .

Similarly, we can see that for WDCS, the values of f for G_1 , G_2 , and G_3 are 0.91, 1.59, and 1.96, respectively. In WDCS, the probability of selecting UEs belonging to G_1 , G_2 , and G_3 is set to 0.16, 0.33, and 0.5, respectively, as the proportion of selecting UEs is set to 1:2:3. In this scheme, the value of f for G_1 and G_2 is almost identical to the corresponding groups in EDCS, while the value of G_3 is lower than that of the corresponding group in EDCS. Despite the higher slot allocation for G_3 UEs in WDCS for packet transmission, the value of f for G_3 in WDCS is lower than the corresponding group in EDCS due to the smaller number of UEs belonging to G_3 .

When we compare the results of RDCS, EDCS, and WDCS in Figure 11, we can say that EDCS is fairer than the other two schemes. If the UEs with medium/bad channel is the priority, then EDCS is the best approach.

4.2.4. Average User Throughput

The average throughput (bandwidth normalized) per user using schemes RDCS, EDCS, and WDCS is shown in Figure 12 for 100 timeslots.

Figure 12. Bandwidth normalized average throughput per user (bps/Hz) for 100 timeslots.

In Figure 12, the average throughput per user for RDCS, EDCS, and WDCS varies from 20.15 (bps/Hz) to 23.52 (bps/Hz), 8.20 (bps/Hz) to 21.96 (bps/Hz), and 4.12 (bps/Hz) to 21.20 (bps/Hz) for different timeslots, respectively. The average throughput per user is higher for RDCS because it mainly selects G_1 candidates, and the UE slot is fully occupied. EDCS and WDCS, on the other hand, reserve slots for UEs from G_2 and G_3 and cause the slot to become free if there is no candidate from those groups in any timeslot.

5. Discussion

In this section, we discuss the results presented in Section 4 and show the limitations of this work.

Section 4.1. presents the performance of the dataset categorization scheme in terms of DLDC model test accuracy and BCE loss. The results presented in Figures 6 and 7 show that the performance of the DLDC model, when trained for 10 training epochs, achieves a test accuracy of 96.7% and a BCE loss of 0.1355. The results presented in Figures 6 and 7 show that the DLDC model is well trained. The trained DLDC model is used to categorize the dataset $\mathcal{D}$ based on G_i in real time. After categorizing the dataset $\mathcal{D}$, the proposed dataset combination procedures were applied.

Section 4.2. presents the performance of dataset combination systems such as RDCS, EDCS, and WDCS is presented based on the complexity analysis, number of users, fairness in user selection, and average user throughput. From the complexity analysis in Section 4.2.1, the complexity of RDCS, EDCS, and WDCS is the same, i.e., $O(TN_u)$. The user selection results shown in Figures 8–10 indicate that the RDCS is likely to fully occupy the slot of UEs since there is no preference level in UE selection, while in the case of the EDCS and WDCS, the full occupancy of the available UE slot is not guaranteed. The result of fairness in UE selection in Figure 11 shows that the EDCS performs better in selecting UEs from groups G_2 and G_3 since it tries to select UEs from each group with the same proportion. Fairness in selecting UEs from G_1 is lower for the EDCS because the number of UEs in G_1 is generally higher. If the number of UEs in each group were the same, the fairness in selecting UEs from each group would be the same under EDCS. The average user throughput is shown in Figure 12. From this, it can be seen that RDCS is the best approach in terms of throughput compared to the other two schemes because it selects

G_1 UEs as they are more in number and there is no preference for selection. Based on the results of fairness in selecting UEs, EDCS is the best approach, while in the case of average system throughput, RDCS is the best approach.

Although EDCS and WDCS provide fairness in selecting the UEs with medium/bad channel. Some of the limitations of the proposed method are:

- In EDCS and WDCS, UEs are selected based on G_i . Therefore, UEs selected under one group for data transmission in one time instance may belong to a different group in another time instance. However, in such a scenario, the UEs that have already had the opportunity to transmit data in the previous time instance may be preferred over the UEs of the same group that have not yet had the opportunity to transmit data.
- If there are no UEs in any of the groups during a time instance under EDCS and WDCS, there will be unused UE slots in those groups during that time instance, which may affect the overall performance of the system.

6. Conclusions

In this paper, a novel dataset categorization and dataset combination method is proposed to ensure a fair selection of UEs for data packet transmission. For dataset categorization, a 2D CNN-based DLDC scheme is proposed to categorize the dataset into three different groups, namely, G_1 , G_2 , and G_3 , based on the radio channel status. The DLDC model was trained with 0.07 M 2D datasets of size 12×3 . The simulation results show that our proposed DLDC model achieves an average test accuracy of 96.7% and BCE loss of 0.1355. Based on the results of the DLDC scheme, it is fed into the systems for combining datasets with schemes such as RDCS, EDCS, and WDCS.

The simulation results for combining datasets show that EDCS is better for selecting UEs of groups G_2 (medium channel capacity) and G_3 (poor channel capacity) because the fairness factor for G_2 and G_3 is comparatively higher in this scheme as it tries to select UEs of groups G_1 , G_2 , and G_3 with the same preference. RDCS, on the other hand, performs better in terms of overall system throughput as it selects most UEs from G_1 (good channel capacity) as they are larger in number. From the results, we can conclude that the dataset generated by EDCS is an ethical dataset since it tries to select UEs from G_1 , G_2 , and G_3 with the same preference.

Author Contributions: Conceptualization, S.B. and H.K.; methodology, S.B.; software, S.B. and N.R.; validation, S.B., N.R., Y.-C.K., P.K. and H.K.; data curation, S.B.; writing—original draft preparation, S.B.; writing—review and editing, S.B., N.R., Y.-C.K., P.K. and H.K.; visualization, S.B., N.R., Y.-C.K. and H.K.; supervision, H.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This work was supported by Post-Doctoral Research Program of Incheon National University in 2017.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Leslie, D. *Understanding Artificial Intelligence Ethics and Safety: A Guide for the Responsible Design and Implementation of AI Systems in the Public Sector*; The Alan Turing Institute: London, UK, 2019.
2. Kearns, M.; Roth, A. *The Ethical Algorithm: The Science of Socially Aware Algorithm Design*; Oxford University Press: New York, NY, USA, 2019.
3. Martin, K. Ethical Implications and Accountability of Algorithms. *J. Bus. Ethics* **2019**, *160*, 835–850. [[CrossRef](#)]
4. Letaief, K.B.; Chen, W.; Shi, Y.; Zhang, J.; Zhang, Y.-J.A. The Roadmap to 6G: AI Empowered Wireless Networks. *IEEE Commun. Mag.* **2019**, *47*, 84–90. [[CrossRef](#)]

5. Kwan, R.; Leung, C.; Zhang, J. Multiuser Scheduling on the Downlink of an LTE Cellular System. *Res. Lett. Commun.* **2008**, *2008*, 323048. [\[CrossRef\]](#)
6. Zulhasnine, M.; Huang, C.; Srinivasan, A. Efficient Resource Allocation for Device-to-Device Communication Underlying LTE Network. In Proceedings of the 2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications, Niagara Falls, ON, Canada, 11–13 October 2010; pp. 368–375.
7. Pokhariyal, A.; Pedersen, K.I.; Monghal, G.; Kovacs, I.Z.; Rosa, C.; Kolding, T.E.; Mogensen, P.E. HARQ Aware Frequency Domain Packet Scheduler with Different Degrees of Fairness for the UTRAN Long Term Evolution. In Proceedings of the 2007 IEEE 65th Vehicular Technology Conference, Dublin, Ireland, 22–25 April 2007; pp. 2761–2765.
8. Ang, E.M.; Wee, K.; Pang, Y.H.; Phang, K.K. A performance analysis on packet scheduling schemes based on an exponential rule for real-time traffic in LTE. *EURASIP J. Wirel. Commun. Netw.* **2015**, *2015*, 201. [\[CrossRef\]](#)
9. Kanhere, S.; Sethu, H.; Parekh, A. Fair and efficient packet scheduling using Elastic Round Robin. *IEEE Trans. Parallel Distrib. Syst.* **2002**, *13*, 324–336. [\[CrossRef\]](#)
10. Kelly, F.P. Charging and Rate Control for Elastic Traffic. *Eur. Trans. Telecommun.* **1997**, *8*, 33–37. [\[CrossRef\]](#)
11. Kim, H.; Han, Y. A proportional fair scheduling for multicarrier transmission systems. *IEEE Commun. Lett.* **2005**, *9*, 210–212.
12. Wang, J.; Chen, X.; Rong, L.; Yiqun, G.; Jun, W. Smart Scheduling Based on Deep Reinforcement Learning for Cellular Networks. In Proceedings of the 2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Helsinki, Finland, 13–16 September 2021; pp. 1–6.
13. Mismar, F.B.; Evans, B.L.; Alkhateeb, A. Deep reinforcement learning for 5G networks: Joint beamforming, power control, and interference coordination. *IEEE Trans. Commun.* **2019**, *68*, 1581–1592. [\[CrossRef\]](#)
14. Bhandari, S.; Kim, H.; Ranjan, N.; Zhao, H.P.; Khan, P. Optimal Cache Resource Allocation Based on Deep Neural Networks for Fog Radio Access Networks. *J. Internet Technol.* **2020**, *21*, 967–975.
15. Sun, Y.; Peng, M.; Zhou, Y.; Huang, Y.; Mao, S. Application of machine learning in wireless networks: Key technologies and open issues. *IEEE Commun. Surv. Tuts.* **2019**, *21*, 3072–3108. [\[CrossRef\]](#)
16. Bhandari, S.; Zhao, H.P.; Kim, H.; Khan, P. Packet Scheduling Using SVM Models in Wireless Communication Networks. *J. Internet Technol.* **2019**, *20*, 1505–1512.
17. Zhang, Z.; Zhai, D.; Zhang, R.; Wang, Y. Deep neural network based channel allocation for interference-limited wireless networks. In Proceedings of the 2019 IEEE High Performance Switching and Routing (HPSR), Xian, China, 26–29 May 2019.
18. Xu, S.; Liu, P.; Wang, R.; Panwar, S.S. Realtime Scheduling and Power Allocation Using Deep Neural Networks. In Proceedings of the 2019 IEEE Wireless Communications and Networking Conference (WCNC), Marrakech, Morocco, 15–18 April 2019; pp. 1–5.
19. Wu, J.; Zhang, G.; Nie, J.; Peng, Y.; Zhang, Y. Deep Reinforcement Learning for Scheduling in an Edge Computing-Based Industrial Internet of Things. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 8017334. [\[CrossRef\]](#)
20. Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data* **2017**, *5*, 153–163. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Bower, A.; Niss, L.; Sun, Y.; Vargo, A. Debiasing representations by removing unwanted variation due to protected attributes. *arXiv* **2018**, arXiv:1807.00461.
22. Agarwal, A.; Dudik, M.; Wu, Z.S. Fair Regression: Quantitative Definitions and Reduction Based Algorithms. In Proceedings of the 2019 International Conference on Machine Learning (ICML), Long Beach, CA, USA, 10–15 June 2019; pp. 120–129.
23. Dai, W.; Yoshigoe, K.; Parsley, W. Improving Data Quality Through Deep Learning and Statistical Models. *arXiv* **2018**, arXiv:1810.07132.
24. Kusner, M.J.; Loftus, J.R.; Russell, C.; Silva, R. Counterfactual Fairness. *arXiv* **2017**, arXiv:1703.06856.
25. Hardt, M.; Price, E.; Srebro, N. Equality of Opportunity in Supervised Learning. In Proceedings of the 2016 Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 3315–3323.
26. Saleiro, P.; Kuester, B.; Hinkson, L.; London, J.; Stevens, A.; Anisfeld, A.; Rodolfa, K.T.; Ghani, R. Aequitas: A Bias and Fairness Audit Toolkit. *arXiv* **2018**, arXiv:1811.05577.
27. Bellamy, R.K.; Dey, K.; Hind, M.; Hoffman, S.C.; Houde, S.; Kannan, K.; Lohia, P.K.; Martino, J.; Mehta, S.; Mojsilovic, A.; et al. AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias. *arXiv* **2018**, arXiv:1810.01943.
28. Larson, J.; Mattu, S.; Kirchner, S.; Angwin, J. Compas Analysis. GitHub. Available online: <https://github.com/propublica/compas-analysis> (accessed on 5 February 2022).
29. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaria, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. [\[CrossRef\]](#)
30. Sultana, F.; Sufian, A.; Dutta, P. Advancements in Image Classification using Convolutional Neural Network. In Proceedings of the 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 22–23 November 2018; pp. 122–129.
31. Chen, Y.; Liu, W.; Niu, Z.; Feng, Z.; Hu, Q.; Jiang, T. Pervasive intelligent endogenous 6G wireless systems: Prospects, theories and key technologies. *Digit. Commun. Netw.* **2020**, *6*, 312–320. [\[CrossRef\]](#)
32. Tariq, F.; Khandaker, M.R.A.; Wong, K.-K.; Imran, M.A.; Bennis, M.; Debbah, M. A Speculative Study on 6G. *IEEE Wirel. Commun.* **2020**, *27*, 118–125. [\[CrossRef\]](#)

-
33. Hancock, J.; Khoshgoftaar, T. Survey on categorical data for neural networks. *J. Big Data* **2020**, *7*, 28. [[CrossRef](#)]
 34. Lofe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:1502.03167.
 35. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 807–814.
 36. Menon, A.K.; Rawat, A.S.; Reddi, S.J.; Kumar, S. Multilabel reductions: What is my loss optimising? In Proceedings of the Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019.
 37. Kingma, D.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
 38. He, S.; Du, J.; Liao, Y. Multi-User Scheduling for 6G V2X Ultra-Massive MIMO System. *Sensors* **2021**, *21*, 6742. [[CrossRef](#)]