

# **Overcoming Data Scarcity through Transfer Learning in CO2-Based Building Occupancy Detection**

Manuel Weber Munich University of Applied Sciences (HM) Munich, Germany Farzan Banihashemi Technical University of Munich Munich, Germany Peter Mandl Munich University of Applied Sciences (HM) Munich, Germany

Hans-Arno Jacobsen University of Toronto Toronto, Canada Ruben Mayer University of Bayreuth Bayreuth, Germany

## ABSTRACT

Knowing indoor occupancy states is crucial for energy optimization in buildings. While neural networks can effectively be used to detect occupancy based on carbon dioxide measurements, their application is impeded by the need for sufficient labeled training data. In this study, we analyze the prediction performance of three different transfer learning (TL) methods leveraging target room data jointly with data from other rooms. The methods include (1) pretraining and fine-tuning, (2) layer freezing, and (3) domain-adversarial learning. Using data from five real-world rooms and one simulated room, including multiple room types, we provide the most extensive evaluation of TL in the field of occupancy prediction from environmental variables to date. This work's contribution further includes the architecture and hyperparameters of a deep CNN-LSTM model for CO2-based occupancy detection. Our results indicate that TL effectively reduces the required amount of target room data. Moreover, while previous literature was focused on pretraining with related real-world data, we show that similar performance can be achieved by the more practical approach of leveraging simulated data.

# **CCS CONCEPTS**

• Computing methodologies  $\rightarrow$  Machine learning; • Humancentered computing  $\rightarrow$  Ubiquitous and mobile computing.

# **KEYWORDS**

Transfer Learning, Deep Learning, CNN-LSTM, Building Occupancy Detection, Indoor Carbon Dioxide

#### **ACM Reference Format:**

Manuel Weber, Farzan Banihashemi, Peter Mandl, Hans-Arno Jacobsen, and Ruben Mayer. 2023. Overcoming Data Scarcity through Transfer Learning in CO2-Based Building Occupancy Detection. In *The 10th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys '23), November 15–16, 2023, Istanbul, Turkey.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3600100.3623718

BuildSys '23, November 15–16, 2023, Istanbul, Turkey © 2023 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0230-3/23/11.

https://doi.org/10.1145/3600100.3623718

#### **1 INTRODUCTION**

Data on building occupancy enables various use cases in terms of building energy performance simulation or building automation [7, 22]. This can have a high impact in the current fight against climate change. As stated by the International Energy Agency (IEA), the buildings sector is responsible for 30% of global energy consumption, with CO<sub>2</sub> emissions annually increasing over the past decade [19]. Moreover, the COVID-19 pandemic has shifted office occupancy away from standardized schedules, which further increases the need for optimization. Integration of occupancy information into heating, ventilation, and air-conditioning (HVAC) systems was experimentally shown to result in energy savings of 37% in [22] and up to 48% in [7]. However, since manual data collection is impractical, especially at the room level, occupancy should be predicted automatically. In [38], the authors outline how occupancy information inferred from environmental sensors can be used for automatic heating set point scheduling. We differentiate between two prediction problems: occupancy detection, which is a binary prediction of presence in general, and occupancy estimation (or counting), the prediction of the number of occupants. In this study, we address the problem of occupancy detection. For the majority of room types, including offices or residential spaces, utilization is limited to a few occupants, and, with respect to energy optimization, exact occupant count information has little advantage over binary states. As the vast majority of today's buildings are not equipped with any dedicated sensing technology, there is a growing branch of research on how to use available technologies or integrate further technology [29, 30]. Technologies explored in the literature include [29], among others, optical or thermal cameras, climate sensors, passive-infrared (PIR) motion sensors, sound sensors, or smart meters to measure energy consumption [11]. Rinta-Homi et al. [27], for example, analyzed occupancy detection based on low-resolution thermal sensors, and in association with this, the trade-off between prediction ability and costs in terms of privacy and deployment. Despite being less accurate than approaches based on camera images, environmental sensing of factors such as carbon dioxide, temperature, or humidity has gained considerable research interest due to its low-cost implementation and low privacy-intrusive nature. Therefore, various machine learning approaches [8, 9, 15, 21, 40], including deep learning [12], have been proposed to infer occupancy from indoor climate. However, scarce data was identified as one of the key challenges for accurate data-driven occupancy prediction [25, 29]. CO<sub>2</sub> rates or other indoor climate factors show

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

different behavior depending on various factors such as room dimensions, infiltration rate, ventilation, or presence of humans or plants. Collecting room-specific labeled training data over large time periods is not applicable in practice. An approach to tackle the impediment of data scarcity is transfer learning (TL). After its huge success in the fields of computer vision and natural language processing, TL has recently made its way into the domain of time series data [33]. In this study, we adopt the definition of TL according to Pan et al. [26]: "Given a source domain  $D_S$  and learning task  $T_S$ , a target domain  $D_T$  and learning task  $T_T$ , transfer learning aims to help improve the learning of the target predictive function  $f_T(\cdot)$  in  $D_T$  using the knowledge in  $D_S$  and  $T_S$ , where  $D_S \neq D_T$ , or  $T_S \neq T_T$ ." We assume that the detection task remains unchanged ( $T_S = T_T$ ), while  $D_S$  and  $D_T$  may refer to different rooms, as in [3, 31], or D<sub>S</sub> may involve simulated data [34, 35]. Recent literature studies [14, 30] reviewing the application of TL methods for occupancy estimation show that, to date, only a few TL-related studies exist for environmental sensing. Moreover, previous works in this context mainly propose a single transfer method and often evaluate it in one specific scenario. This leaves a research gap regarding empirical comparisons of transfer methods on different datasets, which this paper aims to fill. Our contributions include:

- architecture and hyperparameters for a CO<sub>2</sub>-based deep learning occupancy detection model found via an extensive hyperparameter tuning run with data from five rooms,
- the evaluation of three transfer methods, namely pretraining and fine-tuning, layer freezing, and domain-adversarial learning, with six different datasets,
- a comparison between the transfer from similar rooms and a sim-to-real transfer from simulated CO<sub>2</sub> data, and
- the provision of a preprocessed collection of datasets that can be reused in further works on occupancy detection.

## 2 RELATED WORK

Over the last years, several publications have addressed the estimation or binary detection of occupancy based on indoor climate factors including CO2. While some works directly analyze physical dependencies and propose to infer occupancy, for instance, based on the mass balance equation [8], the majority of relevant literature is focused on machine learning (ML) solutions [3, 9, 10, 21], including numerous works leveraging neural networks [12, 18, 40]. ML seems more appropriate in this context since explicit modeling is difficult due to the complexity of indoor CO2 dynamics and other climate factors. Recently, some works have come up that apply TL in the addressed research area in order to improve adaptability to different rooms. Table 1 gives an overview of these. AriefAng et al. [3] initially used a seasonal decomposition model to show that data from an only slightly related domain, in terms of an academic office, can be used to improve prediction performance for a cinema hall. In [2], the authors extended their evaluation by further domains, including classrooms and study zones. Zhang and Ardakanian [39] proposed a transfer between similar rooms in a building using a long short-term memory (LSTM) network. Despite being more generalizable than [2, 3], the approach involves a weighting step relying on domain knowledge. Weber et al. [34, 35] and Stjelja et al. [31] recently applied a deep learning model from [12] to conduct transfer via pretraining and fine-tuning, allowing more generalizability. While [31] focused on similar rooms in a building, [34, 35] proposed the utilization of simulated data to improve predictions for a real-world target office. Another work, by Khalil et al. [23], applied pretraining and fine-tuning with a multi-layer perceptron and a stacked LSTM. With respect to their experimental results on the transfer between offices in a university building, the authors again pointed out the usefulness of TL for occupancy prediction [23]. As well as most of the previous literature, they focus on transfer from similar real-world rooms and do not consider the potential of a

Publication	Problem	Factors	Model	Method	Dataset	Room Type	Dataset Size	Data Rate
Stjelja et al. 2022 [31]	Both	CO <sub>2</sub> , Temp.	Deep NN [12]	Fine-Tuning, Layer Freezing	own <sup>1</sup>	2 Meeting Rooms	19 days, 64 days	3 min
Khalil et al. 2021 [23]	Detect.	CO <sub>2</sub> , Temp., Humidity, Motion	Deep NN (MLP and stacked LSTM)	Fine-Tuning	own <sup>1</sup>	3 Offices	27 months, 3 months, 3 months	5 min
Weber et al. 2020 [34, 35]	Detect.	CO <sub>2</sub>	Deep NN [12]	Fine-Tuning	own <sup>1</sup>	1 Office, 1 Simulated Office	7 days, 500 days	$5 \text{ sec}^2$ , $1 \text{ sec}^2$
Zhang & Ardakanian 2019 [39]	Est.	CO <sub>2</sub> , Air Flow	LSTM, NN	Fine-Tuning with Reweighting	[1], own <sup>1</sup>	2 Study Zones, 2 Classrooms, 3 Meeting Rooms	15 days, 1 year	1 min, 10 min
AriefAng et al. 2018, 2017 [2, 3]	Both	CO <sub>2</sub>	Seasonal Decompo- sition	Domain- Specific	own <sup>1</sup> , [1, 37]	1 Office Room, 1 Cinema Hall, 2 Study Zones, 2 Classrooms	2 months, 45 days, 15 days	5 min, 30 sec, 1 min

#### Table 1: Literature on transfer learning for occupancy prediction from CO<sub>2</sub> and other environmental factors

Detect.: occupancy detection. Est.: occupancy estimation or counting. Temp.: temperature. NN: neural network. MLP: multilayer perceptron. LSTM: long short-term memory.

<sup>1</sup> Data unpublished. <sup>2</sup> Downsampled to 1 min for model training.

Overcoming Data Scarcity through Transfer Learning in CO2-Based Building Occupancy Detection

BuildSys '23, November 15-16, 2023, Istanbul, Turkey

sim-to-real transfer using purely synthetic data from a dedicated simulation. We assume that this can further reduce the need for real-world data. A recent study by Rotem et al. [28] indicates that even pretraining on randomly generated synthetic time series can lead to improvements in time series classification.

While part of the literature on occupancy prediction combines several climate factors [9, 12, 23], such as CO2, temperature, humidity, and air pressure, other works include only CO<sub>2</sub> [8, 18, 20, 21, 36, 40], which was, for instance, identified as the most informative factor in [10]. The  $CO_2$  rate is closely related to human occupancy and generally shows patterns that can clearly be attributed to presence or absence of individuals in a room. Using more input factors can potentially help the model in inconclusive situations but does not necessarily lead to improved overall prediction performance. To the best of our knowledge, there is no broad evaluation showing a clear advantage of using multiple climate factors over using CO<sub>2</sub> only. Kraipeerapun et al. [24] even reported a higher accuracy when using  $CO_2$ , compared to the combination of  $CO_2$  and temperature. In addition to this, even if the same or a slightly higher accuracy could be achieved, temperature is strongly seasonal. This makes an evaluation more difficult, as models trained during the winter season, for instance, may not perform the same on data from a warmer time of the year. Other factors may be more useful. Banihashemi et al. [5] reported a prediction improvement when combining CO<sub>2</sub> with sound pressure level and illuminance. However, while CO<sub>2</sub> sensors are already available in many modern buildings and can be used to optimize ventilation as well, these factors require dedicated sensors. In practice, additional sensor equipment on a room level produces high costs. Aiming at a lower model complexity and better applicability in practice, we use CO<sub>2</sub> time series as single model input. In this setting, TL is particularly important, as it was shown in [5] that, compared to the combination with sound pressure level or illuminance, training only with CO2 requires larger amounts of training data until substantial results are reached.

## **3 METHODOLOGY**

This section introduces the datasets used in this study, the applied deep learning model and transfer methods, as well as the evaluation metrics and the hyperparameter tuning process.

## 3.1 Datasets

In this study, we use  $CO_2$  and occupancy data from multiple realworld datasets of different room types and scenarios, as well as a dataset obtained from simulation.

ID	Name	Room Type	Days	Occ. <sup>1</sup>	PR <sup>2</sup>
А	Office A	office	80	0-2	0.32
В	Office B	office	20	0-2	0.26
С	Home	living & sleeping	50	0-2	0.75
D	Candanedo	office	8	0-2	0.35
Е	Stjelja	meeting room	26	0-12	0.11
F	Simulated	hypothetical office	100	0-1	0.26

**Table 2: Overview of datasets** 

<sup>1</sup>Occ.: typical number of occupants, <sup>2</sup>PR: presence rate.



Figure 1: Layout of the residential room for dataset C.

To ensure comparability, each dataset was resampled to a common 1-min resolution, and occupancy values were binarized to represent presence (1) or absence (0). Incomplete or unoccupied days were removed. Five of the prepared datasets are made publicly available on GitHub.<sup>1</sup> All datasets are summarized in Table 2.

A) Office A: These data were collected at 80 working days between September 2021 and June 2022 in an office room of a university building with about 24 m<sup>2</sup>, located in Munich, Germany. The data were collected through a sensory device and previously used for occupancy detection in a study by Banihashemi et al. [5]. The device consists of multiple sensors to capture indoor environmental qualities (IEQ), including ambient light, sound pressure level, air temperature, indoor air quality, relative humidity, and CO<sub>2</sub>. The data were published under [4]. In this study, we only use the CO<sub>2</sub> data, measured with a Sensirion SCD30 sensor, and the occupancy ground truth, documented by additional manual recording. The room is a two-person office but was predominantly used by a single person due to Covid-19 restrictions.

**B)** Office **B**: Dataset B contains 20 working days from a second office room in another part of the same building used for dataset A. The two rooms have similar dimensions and measurement was conducted equally. The data were as well published under [4].

**C) Home (Living & Sleeping):** For the Home dataset, we collected data from 50 days of occupation in a residential building in Munich, Germany, within the period between 30 April 2020 and 05 July 2020. Due to the Covid-19 lockdown regulations, the room was occupied most of the time and used for working, living, and sleeping. Therefore, this dataset reflects an untypical occupation behavior and contrasts the office datasets with lower occupation rates. The room is located on the ground floor and measures approximately 16 m<sup>2</sup>. CO<sub>2</sub> levels were measured with a Sensirion SCD30 sensor in intervals of 1 sec and later downsampled to a 1-min resolution by averaging. The sensor was placed on top of a bookshelf at a height of approximately 2 m. The room was regularly occupied by two occupants who manually documented their occupancy via button clicking. Figure 1 shows the layout of the room.

<sup>&</sup>lt;sup>1</sup>https://github.com/CCWI/transfer\_learning\_for\_occupancy\_detection/

**D)** Candanedo et al. 2016 (Office): Dataset D contains data measured by Candanedo et al. [9], which were previously used for occupancy detection. It was collected in a two-person office of approximately 20 m<sup>2</sup> within the period between 2nd and 18th of February 2015.  $CO_2$  rates were measured with a Telaire 6613 sensor multiple times per minute and downsampled to a 1-min resolution by averaging. Ground truth was manually inferred from camera pictures. As some days within the data were either not measured completely or fully unoccupied, we selected a subsample of eight complete days. We prepared two further subsamples, in which we additionally removed (D.1) unexpected sensor behavior during non-office hours, and (D.2) the same as (D.1) plus slowly falling  $CO_2$  values during occupation due to opening of the office door.

**E)** Stjelja et al. 2022 (Meeting Room): Dataset E contains data measured by Stjelja et al. [31], which were previously used for occupancy detection. It was collected in a meeting room of a hospital building in Finland. The room measures 21 m<sup>2</sup> and is designed for up to 12 occupants. The measurement was conducted in March and April of 2021. Ground truth was documented in the form of a binary occupation state using an infrared people-counting camera. The authors collected further data, which we did not use, as the room was not operated with constant airflow anymore. From the data collected in [31], we selected a total of 26 occupied days under constant airflow. As the data was available at a 3-min resolution, we applied linear interpolation to obtain 1-min records.

**F) Simulated (Hypothetical Office):** Dataset F contains simulated data for a hypothetical one-person office room with the same dimensions as the room used for dataset A. The data was previously used for occupancy detection by Weber et al. in [34, 35]. First, occupancy was simulated by considering typical office hours for basic status transitions as well as a Markov chain to model random movement. A second Markov chain was used to model the window opening behavior. Afterward, CO<sub>2</sub> rates were determined for each time step based on mass balance calculations. Further details on the simulation can be found in [35]. We selected 100 simulated working days into dataset F.

#### 3.2 Deep Learning Model

The deep learning model used in this study is inspired by the convolutional deep bidirectional long short-term memory (CDBLSTM) proposed by Chen et al. [12]. It was introduced for occupancy estimation based on environmental factors outperforming various previous approaches. Moreover, it has been successfully applied for TL in the context of occupancy prediction [31, 34, 35]. As depicted in Fig. 2, the model combines a one-dimensional convolutional neural network (CNN), for the extraction of local features, and a bidirected long short-term memory (BLSTM), that takes temporal dependencies between these features into account. Subsequently, a feed-forward network of fully connected (FC) layers is used for final classification, with a layer-wise dropout for regularization purposes. The combination of CNN and long short-term memory (LSTM) is a typical approach that is widely applied in TL studies on other time series prediction problems as well [33].

In contrast to [12], we focus on binary occupancy detection, hence, we use sigmoid activation in the output layer instead of softmax. In addition, we treat the number of layers in each of the three model parts as hyperparameters as we assume that the optimal layer numbers depend on the problem setting, including input factors and binary versus multi-class classification. After hyperparameter tuning (see section 3.5), our model consists of two convolutions with 200 and 50 filters, and with kernel sizes of 5 and 3 in the two 1D convolutional layers. Each of the two is followed by a max pooling layer with a pooling size of 2. The convolutional network is connected to three subsequent BLSTM layers with 50 cells per layer. While a stateful BLSTM shows advanced performance if input sequences are not shuffled, we apply a stateless BLSTM and shuffling to avoid overfitting. This is especially required for the domainadversarial model introduced in Sec. 3 to ensure that the domain classifier receives sequences from both domains alternately. The final FC classifier consists of one dropout layer with a dropout rate of 0.5 and one FC layer with 100 neurons. The model is trained with a batch size of 128 and Adam optimizer. We use input sequences of 30 minutes, generated by a sliding window technique on the original time series. To avoid data leakage, we apply sliding windows separately to distinct days in the dataset and use sequences from the same day exclusively for either training or testing.

#### 3.3 Evaluation Metrics

For model evaluation, we use two metrics. One is the accuracy (*acc*) as it is the most widely used evaluation metric in similar studies. It measures the percentage of correct predictions and is given by Eq. 1, where tp, tn, fp, and fn refer to true positive, true negative, false positive, and false negative predictions.

$$acc = \frac{tp + tn}{tp + fp + tn + fn} \tag{1}$$

Since the accuracy is of limited informative value for imbalanced class distributions, it is not well suited, as most rooms, for instance, office rooms, strongly tend to vacancy, especially during the night. Hence, we use Cohen's kappa [13] as a second, more informative metric. Cohen's kappa coefficient ( $\kappa$ ) traditionally measures the agreement between two annotators and can be interpreted as the extent to which a model performs superior over a random classifier. It is calculated according to Eq. 2, where  $p_e$  is the expected accuracy by chance, calculated by p and n, predicted positives and negatives, as well as  $p^*$  and  $n^*$ , positives and negatives in the ground truth.

$$\kappa = \frac{acc - p_e}{1 - p_e}, \qquad p_e = \frac{p \cdot p^* + n \cdot n^*}{(p+n)^2}$$
(2)

#### 3.4 Transfer Learning Methods

In this paper, we apply the following transfer methods, representing the three most widely adopted approaches for model-based time series TL [33].

**Pretraining and Fine-Tuning:** A neural network is pretrained on data from the source domain  $D_S$ , and the trained model parameters are reused to initialize a model for consecutive training, called fine-tuning, with data from the target domain  $D_T$ . Overcoming Data Scarcity through Transfer Learning in CO2-Based Building Occupancy Detection

BuildSys '23, November 15-16, 2023, Istanbul, Turkey



Figure 2: Deep learning model architecture for occupancy detection.

While fine-tuning may include a modification of the training procedure, in this paper, we retrain with training parameters remaining unchanged. We use the last 20% of the source dataset for validation during pretraining. For validation, during fine-tuning, we use one additional target day and apply two variants: In the first, we use only target data, which we call vanilla fine-tuning. In the second, we add an equal amount of source validation data already seen during pretraining.

**Layer freezing:** A common variation of pretraining and finetuning is layer freezing [33]. In this method, fine-tuning on target data only applies to a subset of k out of m model layers  $l_i$ . We finetune the last k layers. Previous layers  $\{l_1, ..., l_n\} \subset \{l_1, l_2, ..., l_m\}$ with n = m - k are frozen, which means their parameters remain unchanged during retraining. We apply two variants, (1) freezing the layers of the CNN part of the model, and (2) freezing both CNN and BLSTM.

**Domain-Adversarial Learning:** This method was introduced by Ganin et al. [17] and uses two competing classifiers, the task classifier (TC) and an additional domain classifier (DC). Both receive features from a preceding feature generator as depicted in Fig. 3. We calculate two loss functions,  $\mathcal{L}_t$  and  $\mathcal{L}_d$ , for TC and DC. In both cases, we apply the binary cross-entropy. A gradient reversal layer (GRL) [17] negates the loss from the domain classification branch. This leads to the following aggregated loss for the feature generator, where  $\lambda$  is a weighting factor that we set to  $\lambda = 1$  in this study:

$$\mathcal{L}_{agg} = \mathcal{L}_t - \lambda \mathcal{L}_d \tag{3}$$

Since the DC tries to discriminate between target and source samples, gradient reversal forces the feature generator to extract domaininvariant feature representations. We apply two variants, placing the DC (1) after the CNN or (2) after the BLSTM.



Figure 3: Domain-Adversarial Architecture. GRL: Gradient Reversal Layer.

## 3.5 Hyperparameter Tuning

To find the model architecture and training parameters described in Sec. 3.2, we conducted a hyperparameter tuning process divided into two phases: In phase 1, the task classifier, i.e., the base CDBLSTM, was tuned. In phase 2, the domain classifier of the domain-adversarial model was tuned, reusing task classifier parameters from phase 1.

#### Table 3: Hyperparameters, cardinality (#), and search values

Hyperparameter	#	Search Values
A. Training Parameters:		
Batch Size	3	[32, 64, 128]
Window Size	3	[15, 30, 60]
Optimizer	4	[SGD, RMSprop,
		Adam, Adadelta]
B. Number of Layers:		
Number of Convolutions		[1, 2]
Number of BLSTM Layers		[1, 2, 3, 4]
Number of Dense Layers	4	[1, 2, 3, 4]
Number of Layers in DC <sup>3</sup>	3	[0, 1, 2]
C. Further Hyperparamters:		
Stateful BLSTM	2	[True, False]
Convolutional Filters <sup>1</sup>		[50, 100, 150, 200]
Convolution Kernel Size <sup>1</sup>		[3, 5]
Number of BLSTM Cells <sup>12</sup>		[50, 100, 150,, 500]
Number of Dense Neurons <sup>1 2</sup>		[100, 200,, 500]
Number of Neurons in DC <sup>1 2 3</sup>		[10, 20,, 500]
Dropout Rates <sup>1 2</sup>		[0.1, 0.3, 0.5]
Dropout Rates in DC <sup>1 2 3</sup>		[0.1, 0.3, 0.5]

<sup>1</sup> May exist multiple times, depending on the selected number of layers (B.), with possibly different values each.

<sup>2</sup> The listed values apply to the first layer. Consecutive layers are restricted to have the same or half the value of the previous (rounded up to the next listed discrete value), or the minimum.

<sup>3</sup> Refers to the domain classifier (DC) tuned separately in phase 2.

Phase 1: Base model tuning was conducted by applying a Bayesian optimization (BO). BO is an informed, sequential optimization strategy based on the Bayes theorem that selects promising parameter combinations according to previous evaluations. It is well-suited for computationally expensive evaluations and outperforms uninformed strategies such as grid search or random search [32]. We conducted 1000 iterations of BO to approximate the optimal hyperparameter setting within the search space, and used all real-world datasets (A-E) for the tuning procedure. For each dataset, we reserved the first 80% of the data for hyperparameter tuning, the remaining 20% were held out for testing purposes. Among the first 80%, we randomly selected 5 consecutive days of data for training and the following 3 days for validation. We decided on a training data amount of 5 days, as our preliminary examinations showed this to be the minimum amount of data needed for the untuned deep learning model to produce solid results. By keeping the training data amount small, we ensure fast training times and allow a large number of parameter settings to be searched in the tuning procedure. Table 3 lists all parameters and their considered values included in the hyperparameter search. We included (A) parameters related to the training process and input data, i.e., different batch sizes and optimizers, as well as input window sizes of either 15-, 30-, or 60-minute CO<sub>2</sub> sequences, (B) different numbers of layers for all model components, and (C) further hyperparameters such as the number of neurons, filters, and BLSTM cells. In the case of multiple layers per model component, we reduced the complexity of the search space by constraining the number of neurons or cells in consecutive layers to either (1) the minimum allowed value, (2) the same as the preceding layer, or (3) half of the preceding layer's value rounded up to the next allowed value.

**Phase 2:** In phase 2, we tuned the number of layers and neurons as well as dropout rates in the domain classifier branched from the base model for domain-adversarial learning. We applied a BO with 100 iterations. Within each iteration, we tested each of 20

possible source-to-target combinations within the set of datasets A-E, with 5 repetitions each, and calculated the average Cohen's kappa accordingly. While again using five days of training data from the source dataset, we used one additional day from the target dataset for training and three target days for validation.

#### 3.6 Experimental Setup

We applied a min-max normalization separately to each training, validation, or test split. Model training and interference were conducted on an Nvidia GeForce Tesla V100 SXM2. Keras and Tensorflow were used for implementation. All training procedures were run with early stopping after five epochs without loss improvement. The source code used for the experiments is made publicly available on GitHub.<sup>2</sup>

## 4 RESULTS

To analyze transfer performance, we conducted multiple experiments applying each of the transfer methods introduced in Sec. 3.4 to each possible combination of source datasets within A-F and distinct target datasets within A-E. Simulated data (F) was not used as target, as this would not represent any real-world use case. Each experiment was repeated 20 times to increase reliability.

Fig. 4 shows the results for an exemplary selection of experiments where two days of target data were used. The results indicate that depending on the specific data, different transfer methods result in remarkable differences in prediction performance. While pretraining and fine-tuning, for instance, performs best for combinations 3 and 5, domain-adversarial learning shows superior results in case 4 (Stjelja to Candanedo) and layer-freezing in case 1 (office B to A). In addition to this, there are also differences in reliability. Especially layer-freezing often shows large standard deviations.



<sup>2</sup>https://github.com/CCWI/transfer\_learning\_for\_occupancy\_detection/

Figure 4: Mean and standard deviation of Cohen's kappa and accuracy by transfer method for five selected experiments with two days of target data. The highest mean values are highlighted in red. t: target-only training (no transfer), PF: pretraining and vanilla fine-tuning, PF2: pretraining and fine-tuning with source and target data, LF: layer freezing (frozen CNN), LF2: layer freezing (frozen CNN-BLSTM), DA1/DA2: domain-adversarial training variant 1 or 2.

In our experiments, the best transfer method almost always performs superior to non-transfer, regardless of the source dataset, except for some exceptions targeting the meeting room dataset (E) by Stjelja et al. Combinations with dataset E suffer the most from negative transfer. This may be due to the data interpolation from 3-min to 1-min resolution, which was only necessary for this dataset, or to the room type being the most dissimilar compared to the other datasets. All other rooms are regularly used by 1-2 occupants, whereas for E, the room is mostly unoccupied and there are up to 12 occupants. Also, transfer to E shows low benefit, as the model can already reach substantial results with scarce training data only. When using two days of data from E, for example, in comparison with training from scratch, pretraining on A only increases Cohen's kappa from 0.648 to 0.659 and accuracy from 0.938 to 0.939. For other target datasets, however, training a model from scratch with such scarce data is not possible, which can be seen from accuracies around 0.6 and kappa values close to zero. This offers large potential for TL.

## 4.1 Analysis of Transfer Methods



Figure 5: Mean Cohen's kappa and accuracy by target training days and transfer method over all source-to-target combinations between sources A-F and targets A-D.

Fig. 5 shows the mean performances over all source-to-target combinations with target datasets A-D. It can be seen that domainadversarial learning shows remarkable performance in terms of Cohen's kappa in cases with extremely scarce data of only one target day. In comparison to the other methods, however, it appears to fail in making use of more target data. In our evaluation, for multiple training days, the mean accuracy remains close to the baseline accuracy without transfer learning, and below the accuracy of other transfer methods. Pretraining and fine-tuning stands out as the best method when averaging over the experiments. Especially when fine-tuning is conducted jointly with data from target and source domain, there is a further performance advantage. The approach intends to reduce catastrophic forgetting during fine-tuning with scarce data by repeatedly training with data previously seen during pretraining. Regarding the results shown in Fig. 5, it seems that this is particularly useful in cases with extremely scarce target data. With three target days or more, the performance of vanilla fine-tuning approaches a similar level, although still performing weaker. It shall be noted that target performance also depends on the selected source data. A comparison of source datasets is addressed in subsection 4.2. For specific sources, methods can reach a higher or lower performance than the mean values reported in Fig. 5. To show this, Fig. 6 provides a more detailed overview showing the mean Cohen's kappa values of the methods for each source dataset for one and five target training days respectively. Even with extremely scarce target data of only one day, most of the methodsource combinations show significantly higher performance than the baseline without transfer, which is close to zero and cannot be regarded as a useful model. With five days, performance further increases in most cases. The top performance is reached on source dataset C by pretraining and fine-tuning with source and target data. Negative transfer can be observed in some cases, mostly with datasets D or E and with layer freezing or domain-adversarial learning. Hence, these methods seem less advisable when datasets are too diverse. Pretraining and fine-tuning appears more flexible. With five target training days, it shows positive transfer for each of the sources. This can be explained by the necessity of early layers being retrained when there are massive pattern changes, which is omitted in the layer freezing method.

## 4.2 Analysis of Transfer Sources

Fig. 7 shows the mean Cohen's kappa and accuracy values grouped by the source datasets for 1-5 days of target training data. Dataset E was excluded as target dataset in this evaluation to avoid negative transfer affecting the results. For each source, values are reported for the best transfer method, as we suggest that in each case the best method would be applied in practice. A more detailed overview regarding the combination of sources and methods is shown in Fig. 6. The results show a clear dominance of transfer learning over targetonly training for all source datasets. Datasets D and E perform weaker than other sources. This may again be due to major dissimilarities to the other datasets. The two offices A and B perform similarly. For more than one target day, the simulated data show similar results. This indicates that it is possible to successfully apply TL even without extensive real-world data collected from other rooms.



Figure 6: Mean Cohen's kappa over target datasets A-D by source dataset and transfer method for 1 day (left) or 5 days (right) of target data used for training. t: target-only training (no transfer), PF: pretraining and vanilla fine-tuning, PF2: pretraining and fine-tuning with source and target data, LF: layer freezing (frozen CNN), LF2: layer freezing (frozen CNN-BLSTM), DA1/DA2: domain-adversarial training variant 1 or 2.



Figure 7: Mean Cohen's kappa and accuracy by target training days and source dataset for the best transfer method in each source-to-target combination between sources A-F and targets A-D.



Figure 8: Mean Cohen's kappa by target training days and source dataset for the best transfer method in each experiment with sources B-F and office A as the target. Results for two selected subsets of D are shown in dashed lines.

Weber et al.

Overcoming Data Scarcity through Transfer Learning in CO2-Based Building Occupancy Detection

The home dataset (C) stands out as the best source by average. With its high presence rate of 75% and dataset size of 50 days, it provides the most instances from the positive class. However, regarding a specific target room and with little target data, other sources may be more suitable, as can be seen in Fig. 8.

Fig. 8 shows the kappa values for different sources when office A is the target dataset. As expected, the most similar room, office B, appears to be the most suitable real-world source. However, with more than two target days, the simulated dataset F performs on par with office B.

Dataset D shows noticeable poor performance in transfer to A, although it was measured in a two-person office as well. Closer investigation showed some peculiar CO2 measurements during non-office hours that, according to the authors, cannot be attributed to any known real-world phenomenon. Removing these from the data increases performance to some extent (see D.1). A further regular scenario that differs from the other datasets involves slowly decaying CO<sub>2</sub> values despite occupancy. The authors explain this by opening of the office door to let fresh air in. In dataset A, however, ventilation took place by window opening. This causes a fast drop of CO<sub>2</sub> that is more distinctive from vacancy. Removing also these unique decay scenarios (see D.2), performance becomes more comparable to other sources. This indicates that, besides the similarity of rooms, also the particularities of their usage are relevant regarding transfer capabilities. Despite these difficulties, the transfer from dataset D to A, just as from all other source datasets, improves the results to an important degree compared to training without transfer learning.

## 5 DISCUSSION

Our results show that TL broadly benefits occupancy detection based on  $CO_2$  in many different scenarios. While this study is focused on the problem of scarce data and, hence, reports results for up to five working days in the target room, we observed further prediction improvements with larger amounts of target data as well. For instance, when using 40 working days from dataset A for target training, we still observed a slight advantage when using a model pretrained on 40 days of the simulated dataset D, with a Cohen's kappa of 0.8879 compared to 0.8808. This may motivate the use of TL even in cases where data scarcity is not a major problem, although it can be assumed that the performance advantage of TL tends towards zero with increasing target dataset size.

However, since data is often scarce in practice, TL may play a key role in making the approach applicable. A remaining difficulty is the observed performance variance depending on the concrete source dataset being used. Dissimilarities between source and target domain may appear due to differences in

- 1. properties of the room (size, ventilation type, etc.), or
- 2. occupant behavior (e.g., door or window opening behavior).

Indoor climate, for example, fundamentally differs between rooms with air-conditioning versus natural ventilation. While the applied methods are transferable, in this study, we focused on naturally ventilated rooms. A dissimilarity caused by occupant behavior found in this study was due to occupants using the office door instead of the window to ventilate the room. Dissimilarities leading to negative transfer should be prevented in practice, which is why selecting the right source dataset and transfer method is crucial. Approaches on how to prevent negative transfer may be:

- Selecting appropriate sources, either by actually testing the prediction performance or based on estimation. Regarding TL for time series classification in general, Fawaz et al. [16] suggested calculating a dynamic time warping (DTW) distance as a proxy for dataset similarity.
- 2. Selecting a transfer method that reduces the risk of negative transfer. In our evaluation, pretraining and fine-tuning showed less tendency to negative transfer. In contrast, freezing layers during retraining carries the risk that parameters representing patterns deviating from the target domain cannot be overwritten.
- **3.** Multi-source training: Leveraging multiple source datasets at once, in order to train a more general model, that is less overfitting to a specific source room or occupant behavior.

Regarding multi-source training, however, it is a challenge to collect sufficient source data, as public datasets are rare. To avoid the restriction of limited real-world data being available, we propose leveraging simulated data instead. As shown in our evaluation, there is no discernable discrepancy between simulated source data and real-world source data. When transferring from the simulated dataset with more than two days of target data, we observed performances similar to the top-performing real-world datasets (cf. Fig. 7 and Fig. 8). Hence, we point out that the application of simulations in occupancy detection modeling needs to be further addressed in future research. Two alternative ideas were introduced in [34], which are (1) to replicate the conditions in the target room as closely as possible or (2) to simulate under a broad variety of conditions to generate data for multi-source training. The first option requires a large amount of manual work, including the tasks of room modeling, simulating, data preprocessing, and two phases of model training, whereas the latter may allow the preparation of a more general base model that can quickly be applied by fine-tuning on any given real-world data. This further reduces human involvement compared to collecting suitable real-world datasets for TL. As an alternative future research direction, we suggest to also compare the approach to unsupervised learning algorithms as, for instance, investigated in the context of energy consumption data in [6].

## **6** CONCLUSIONS

In this paper, we proposed a model architecture and configuration for the task of building occupancy detection from CO<sub>2</sub>, based on an existing deep learning approach. We evaluated the model's capability of transfer learning, comparing three different transfer methods on a variety of datasets. The results suggest that pretraining and fine-tuning is generally a good choice in the addressed domain of application, even though it is not necessarily the best method in all cases. According to our experiments, its performance can be increased by fine-tuning jointly with data from target and source domain. In addition, in experiments with sufficient finetuning, pretraining with simulated data performed on par with using real-world data from a similar room. This finding underlines the usefulness of simulations in practice, as it allows us to train models with minimal data collection. In future work, we intend to pretrain an off-the-shelf model based on a variety of simulations.

## ACKNOWLEDGMENTS

We would like to thank Davor Stjelja from Aalto University in Finland for kindly providing us with the meeting room data. Further thanks go to IBM and the *OpenPower@TUM* project for providing us with the GPU computing infrastructure.

#### REFERENCES

- [1] Krzysztof Arendt, Aslak Johansen, Bo Nørregaard Jørgensen, Mikkel Baun Kjærgaard, Claudio Giovanni Mattera, Fisayo Caleb Sangogboye, Jens Hjort Schwee, and Christian T. Veje. 2018. Room-level occupant counts, airflow and CO 2 data from an office building. *First Workshop on Data Acquisition To Analysis (DATA)* (2018), 13–14.
- [2] Irvan B. Arief-Ang, Margaret Hamilton, and Flora D. Salim. 2018. A Scalable Room Occupancy Prediction with Transferable Time Series Decomposition of CO2 Sensor Data. ACM Transactions on Sensor Networks 14, 3-4 (2018), 1–28.
- [3] Irvan B. Arief-Ang, Flora D. Salim, and Margaret Hamilton. 2017. DA-HOC: Semi-Supervised Domain Adaptation for Room Occupancy Prediction using CO2 Sensor Data. Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys) (2017), 1–10.
- [4] Farzan Banihashemi and Manuel Weber. 2023. "Non-intrusive indoor environmental data in two double-occupied offices". *Mendeley Data* (2023). https: //doi.org/10.17632/v7kw9ycccp.1
- [5] Farzan Banihashemi, Manuel Weber, Fatma Deghim, Chujun Zong, and Werner Lang. 2023. Occupancy modeling on non-intrusive indoor environmental data through machine learning. SSRN (2023). https://doi.org/10.2139/ssrn.4477794
- [6] Vincent Becker and Wilhelm Kleiminger. 2018. Exploring zero-training algorithms for occupancy detection based on smart meter measurements. *Computer Science - Research and Development* 33, 1 (2018), 25–36.
- [7] Jonathan Brooks and Prabir Barooah. 2014. Energy-efficient control of underactuated HVAC zones in buildings. 2014 American Control Conference (2014), 424–429.
- [8] Davide Calì, Peter Matthes, Kristian Huchtemann, Rita Streblow, and Dirk Müller. 2015. CO2 based occupancy detection algorithm: Experimental analysis and validation for office and residential buildings. *Building and Environment* 86 (2015), 39–49.
- [9] Luis M. Candanedo and Véronique Feldheim. 2016. Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models. *Energy and Buildings* 112 (2016), 28–39.
- [10] Luis M. Candanedo, Véronique Feldheim, and Dominique Deramaix. 2017. A methodology based on Hidden Markov Models for occupancy detection and a case study in a low energy residential building. *Energy and Buildings* 148 (2017), 327–341.
- [11] Dong Chen, Sean Barker, Adarsh Subbaswamy, David Irwin, and Prashant Shenoy. 2013. Non-Intrusive Occupancy Monitoring Using Smart Meters. Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings (BuildSys) (2013), 1–8.
- [12] Zhenghua Chen, Rui Zhao, Qingchang Zhu, Mustafa K. Masood, Yeng Chai Soh, and Kezhi Mao. 2017. Building Occupancy Estimation with Environmental Sensors via CDBLSTM. *IEEE Transactions on Industrial Electronics* 64, 12 (2017), 9549–9559.
- [13] Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. Educational and Psychological Measurement 20, 1 (1960), 37–46.
- [14] Jawher Dridi, Manar Amayri, and Nizar Bouguila. 2022. Transfer learning for estimating occupancy and recognizing activities in smart buildings. *Building and Environment* 217 (2022), 109057.
- [15] Afrooz Ebadat, Giulio Bottegal, Damiano Varagnolo, Bo Wahlberg, and Karl H. Johansson. 2013. Estimation of Building Occupancy Levels through Environmental Signals Deconvolution. Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings (BuildSys) (2013), 1–8.
- [16] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. 2018. Transfer learning for time series classification. *IEEE International Conference on Big Data* (2018), 1367–1376.
- [17] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research* 17 (2016), 1–35.
- [18] Hwataik Han, Kyung-jin Jang, Changho Han, and Junyong Lee. 2013. Occupancy estimation based on CO2 concentration using dynamic neural network model. *Proc. 34th Air Infiltration and Ventilation Centre Conference (AIVC)* (2013).

- [19] IEA. 2022. World Energy Outlook 2022. https://www.iea.org/reports/worldenergy-outlook-2022
  [20] Chaoyang Jiang, Zhenghua Chen, Rong Su, Mustafa Khalid Masood, and
- [20] Chaoyang Jiang, Zhenghua Chen, Rong Su, Mustafa Khalid Masood, and Yeng Chai Soh. 2020. Bayesian filtering for building occupancy estimation from carbon dioxide concentration. *Energy and Buildings* 206 (2020), 109566.
- [21] Ming Jin, Nikolaos Bekiaris-Liberis, Kevin Weekly, Costas J. Spanos, and Alexandre M. Bayen. 2018. Occupancy Detection via Environmental Sensing. *IEEE Transactions on Automation Science and Engineering* 15, 2 (2018), 443–455.
- [22] Jonathan Brooks, Siddharth Goyal, Rahul Subramany, Yashen Lin, Timothy Middelkoop, Laura Arpan, Luca Carloni, and Prabir Barooah. 2014. An experimental investigation of occupancy-based energy-efficient control of commercial building indoor climate. IEEE Conference on Decision and Control 53 (2014), 5680–5685.
- [23] Mohamad Khalil, Stephen McGough, Zoya Pourmirza, Mehdi Pazhoohesh, and Sara Walker. 2021. Transfer Learning Approach for Occupancy Prediction in Smart Buildings. 12th International Renewable Engineering Conference (IREC) (2021), 1–6.
- [24] Pawalai Kraipeerapun and Somkid Amornsamankul. 2017. Room Occupancy Detection using Modified Stacking. Proceedings of the 9th International Conference on Machine Learning and Computing (ICMLC) (2017), 162–166.
- [25] William O'Brien, Andreas Wagner, Marcel Schweiker, Ardeshir Mahdavi, Julia Day, Mikkel Baun Kjærgaard, Salvatore Carlucci, Bing Dong, Farhang Tahmasebi, Da Yan, Tianzhen Hong, H. Burak Gunay, Zoltan Nagy, Clayton Miller, and Christiane Berger. 2020. Introducing IEA EBC annex 79: Key challenges and opportunities in the field of occupant-centric building design and operation. Building and Environment 178 (2020), 106738.
- [26] Sinno Jialin Pan and Qiang Yang. 2010. A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering 22, 10 (2010), 1345–1359.
- [27] Mikko Rinta-Homi, Naser Hossein Motlagh, Agustin Zuniga, Huber Flores, and Petteri Nurmi. 2021. How Low Can You Go? Performance Trade-Offs in Low-Resolution Thermal Sensors for Occupancy Detection: A Systematic Evaluation. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. 5, 3 (2021).
- [28] Yarden Rotem, Nathaniel Shimoni, Lior Rokach, and Bracha Shapira. 2022. Transfer Learning for Time Series Classification Using Synthetic Data Generation. *Cyber Security, Cryptology, and Machine Learning (CSCML): 6th International Symposium (2022), 232–246.*
- [29] Luis Rueda, Kodjo Agbossou, Alben Cardenas, Nilson Henao, and Sousso Kelouwani. 2020. A comprehensive review of approaches to building occupancy detection. *Building and Environment* 180 (2020), 106966.
- [30] Aya Nabil Sayed, Yassine Himeur, and Faycal Bensaali. 2022. Deep and transfer learning for building occupancy detection: A review and comparative analysis. *Engineering Applications of Artificial Intelligence* 115 (2022), 105254.
- [31] Davor Stjelja, Juha Jokisalo, and Risto Kosonen. 2022. Scalable Room Occupancy Prediction with Deep Transfer Learning Using Indoor Climate Sensor. *Energies* 15, 6 (2022).
- [32] Ryan Turner, David Eriksson, Michael McCourt, Juha Kiili, Eero Laaksonen, Zhen Xu, and Isabelle Guyon. 2021. Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020. Proc. NeurIPS 2020, PMLR 133 (2021), 3–26.
- [33] Manuel Weber, Maximilian Auch, Christoph Doblander, Peter Mandl, and Hans-Arno Jacobsen. 2021. Transfer Learning With Time Series Data: A Systematic Mapping Study. IEEE Access 9 (2021), 165409–165432.
- [34] Manuel Weber, Christoph Doblander, and Peter Mandl. 2020. Detecting Building Occupancy with Synthetic Environmental Data. Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (BuildSys) (2020), 324–325.
- [35] Manuel Weber, Christoph Doblander, and Peter Mandl. 2020. Towards the Detection of Building Occupancy with Synthetic Environmental Data: arXiv:2010.04209.
- [36] Yixuan Wei, Shu Wang, Longzhe Jin, Yifei Xu, and Tianqi Ding. 2022. Indoor occupancy estimation from carbon dioxide concentration using parameter estimation algorithms. *Building Services Engineering Research and Technology* 43, 4 (2022), 419–438.
- [37] Jörg Wicker, Nicolas Krauter, Bettina Derstorff, Christof Stönner, Efstratios Bourtsoukidis, Thomas Klüpfel, Jonathan Williams, and Stefan Kramer. 2015. Cinema Data Mining: The Smell of Fear. Proc. 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2015), 1295–1304.
- [38] Dominic Wörner, Thomas von Bomhard, and Felix Wortmann. 2013. Occupancy-Based Heating Control for Residential Buildings Using Environmental Sensors. Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings (BuildSys) (2013), 1-2.
- [39] Tianyu Zhang and Omid Ardakanian. 2019. A domain adaptation technique for fine-grained occupancy estimation in commercial buildings. Proc. 2019 Internet of Things Design and Implementation (IoTDI) (2019), 148-159.
- [40] M. S. Zuraimi, A. Pantazaras, K. A. Chaturvedi, J. J. Yang, K. W. Tham, and S. E. Lee. 2017. Predicting occupancy counts using physical and statistical Co2-based modeling methodologies. *Building and Environment* 123 (2017), 517–528.