Digital Twin Development for Stack Effect Mitigation

Gary Chang, gary.chang@ryerson.ca
Department of Architectural Science, Ryerson University, Toronto, Canada

Karim El Mokhtari, elmkarim@ryerson.ca
Department of Architectural Science, Ryerson University, Toronto, Canada

Russell Richman, richman@ryerson.ca
Department of Architectural Science, Ryerson University, Toronto, Canada

J. J. McArthur, jennifer.mcarthur@ryerson.ca
Department of Architectural Science, Ryerson University, Toronto, Canada

Abstract
Stack effect adversely affects heating loads in high-rise buildings due to the excess infiltration caused by the pressure differentials. Mitigating the effects phenomenon is crucial to achieve ambitious energy targets such as Canada's commitment to COP21. A novel data-driven modelling approach is leveraged to predict if stack effect is problematic based on the weather for the case study building. The study investigates machine learning strategies by applying classification metrics to determine the most suited for stack effect modelling. It also provides the framework for a digital twin to predict and mitigate stack effect, serving as the foundation for the development of recommender algorithm, which will use the classification within the DT to test a range of potential control strategies and recommend the optimal approach (based on minimum energy cost to achieve stack effect control) to facility management staff.

Keywords: Stack Effect, Machine Learning, BIM

1 Introduction
Stack effect is the pressure differential and resultant rapid upward air movement caused by natural convection (Wilson & Tamura, 1968). In conventional high-rise buildings, there are openings at the top (vents) and bottom (entrance), which allow cold (winter) or warm (summer) air to enter the building. The warmer air rises above the cold, resulting in a linear pressure differential distribution, as shown in Figure 1.

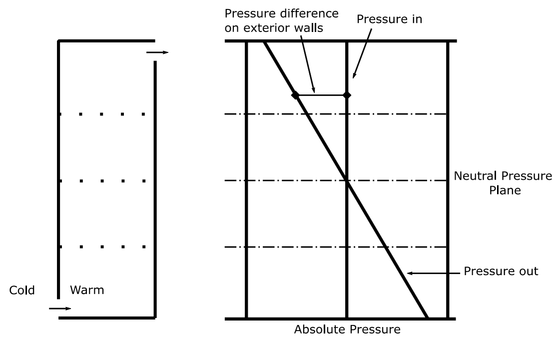

Figure 1 Stack Effect in Heating Season Visualization based on Wilson & Tamura **(1968)**

In the heating season, this causes increased air infiltration on levels below the neutral pressure plane, and increased air exfiltration above it. During the cooling season, the effect is reversed and may be beneficial in mild climates as it can drive natural (passive) ventilation (Yu, et al., 2017). Stack effect is generally undesirable, however, and is estimated to contribute to 10.27% of the heating load in high-rise residential buildings (Yoon, et al., 2019), making its mitigation a significant consideration for energy conservation. Occupant health and comfort is another key concern as the additional air movement due to stack effect causes unwanted noise in elevator shafts (Lovatt & Wilson, 1994), movement of pollutants such as airborne viruses (Lim, et al., 2011), and smoke (Zhang, et al., 2006), alongside potential thermal comfort issues.

Stack effect has been traditionally difficult to model. Physics-based models have been used to simulate stack effect in case study buildings to investigate potential countermeasures (Lim, et al., 2020) but are limited by: the need to perform large scale testing to acquire data such as air leakage, the large computational processing power required, and the resultant model complexity. With the application of data-driven techniques to other aspects of building performance, we believe there is promise in such an approach, however there is a paucity of such studies. This paper thus makes a novel contribution of a *grey-box* approach leveraging machine learning alongside simplified physics models to predict stack effect. This builds on a previous grey-box model (G1) that used return air temperature as a proxy for thermal comfort issues arising from stack effect (Chang, et al., 2021), this time considering temperature differences between the duct-mounted return air temperature and corresponding average floor temperatures. This new model (G2) can thus identify high rates of air infiltration and uses this for stack effect prediction. In this paper, it is compared to G1, which was updated to extend its domain to the shoulder season. Data was gathered from a case study building: a high-rise office tower in Toronto, Canada, for which a Digital Twin (DT) is being developed. In the next phase of research, the stack effect model will be validated with data from field-installed differential pressure sensors and will be integrated into a recommender algorithm, which will test a series of potential control strategies for a given weather forecast and set of operating conditions to control the stack effect at a minimal energy and carbon cost. This project is being conducted in collaboration with facility management staff to ensure that the DT will provide actionable information.

2 Literature Review

Stack effect modeling follows the same principles as airflow modeling as they are fundamentally similar. Based on the work of Chen (2009), stack effect modeling methods can be distilled as follows: multi-zone model, computational fluid dynamics (CFD), and coupled models. The multi-zone performs airflow analysis by treating the building as a series of nodes or zones that are interconnected, along with boundary conditions and air properties to simulate the airflow (Axley, 2007). Yu et al. (2017) leveraged the multi-zone model to simulate the effect of stack effect in their high-rise office tower, and ran various pressurization schemes to determine the most effective mitigation strategy. CFD is provides more accurate results when compared to the multi-zone model due to less simplifications in the parameters, but this leads to higher computational power requirements (Chen, 2009). Wong and Heryanto (2004) studied opportunities to enhance natural ventilation with stack effect by implementing a CFD model, but was limited to one apartment unit due to the high computational demands. Coupled models combine the multi-zone method with data-driven modeling, in order to utilize their respective strengths and compensate the weaknesses. Yoon et al. (2015) utilized the coupled method to model a multi-residential case study building, where the multi-zone was used as the base model, and unknown parameters such as air leakage areas was inferred through genetic algorithms. The resultant model produces more accurate results compared to the multi-zone model but significantly raises model complexity. When modeling stack effect, key parameters to consider are the airtightness of the building, zone temperature, zone volume, air leakage areas, weather data, and the HVAC characteristics (Yoon, et al., 2015).

A validated model to simulate stack effect in a case study building allows various mitigation strategies to be tested to evaluate their impact. Lim et al. (2020) performed a comprehensive study on available stack effect countermeasures and their interactions. They divided the countermeasures into architectural solutions and mechanical solutions. Architectural solutions focus on changing the pressure differential profile of the building by improving envelope air tightness, adding interior partitions, or otherwise compartmentalizing the building (Lim, et al., 2020). These solutions are effective in reducing the potential air leakage area that allows for uncontrolled air movement (Yu, et al., 2004). Mechanical solutions utilize the HVAC system to pressurize spaces within the building to mitigate the pressure differentials causing to stack effect. Mechanical solutions are more feasible to implement as they don't require costly retrofitting of the building envelope (Tamblyn, 1991), however, the correct pressurization scheme has to be identified for each case study building as the wrong approach can amplify stack effect in unpressurized spaces (Lovatt & Wilson, 1994).

Data-driven modeling is rapidly rising in popularity in research as they allow for the creation of accurate models, while reducing model complexity and computational cost compared to physics-based models. Data-driven modeling is also known as machine learning, is split into four categories: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning (Molina-Solana, et al., 2017). In the context of this research, supervised learning methods were the most relevant as they map the inputs to the outputs based on a given dataset to develop a relationship between these parameters from statistical learning. There are numerous statistical learning algorithms to choose from, with each using a fundamentally different approach in building a model.

Despite the value of data-driven modeling, a research gap still exists with respect to applying this approach to developing a stack effect model. As a result, this study had no direct methodological precedents, and thus broader building system fault detection and diagnosis (FDD) studies were reviewed to inform algorithm selection. Five potential algorithms were thus identified: Linear Discriminate Analysis (LDA), which has been used to classify faults in chiller operation (Li, et al., 2016); K-Nearest Neighbor (KNN), used to successfully diagnose multiple fault classes for solar PV systems (Madeti & Singh, 2018); Random Forest (RF) and Support Vector Machines (SVM), which both provided very good results for non-condensing boiler faults (Shohet, et al., 2020); and Multi-Level-Perceptron (MLP), also known as a feed forward approach for artificial neural networks, used fault detection of energy consumption in a cluster of smart office buildings (Capozzoli, et al., 2015).

3 Methodology

The study adopted the traditional method of creating a predictive data-driven model of stack effect, which was integrated with BIM visualizations for use as a facilities management tool. Figure 2 illustrates the workflow for the G2 model. The steps have been split as follows: (1) Data Collection; (2) Data Visualization and Pre-processing; (3) Model Training; (4) Model Validation and Recalibration; (5) BIM Visualization and Integration

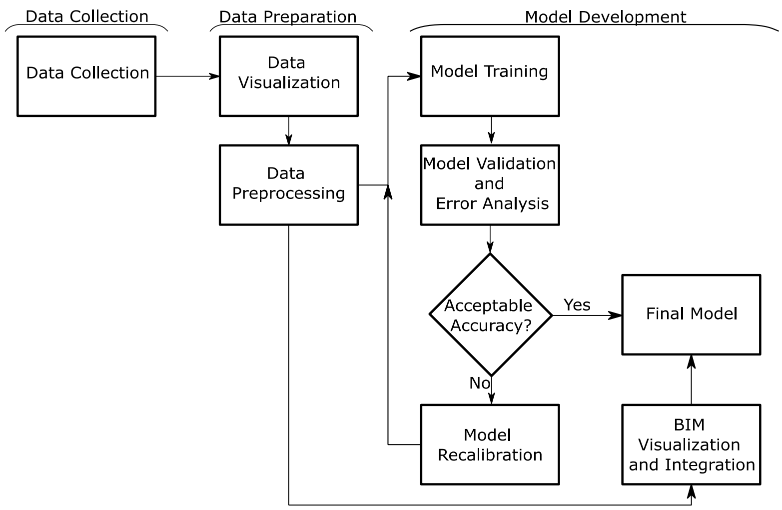

Figure 2 Methodology Flow Chart

3.1 Data Collection

Data was collected from a 223m case study building: a high-rise office tower 223m tall, with 56 above-grade floors and a total area of 123,422 m², located in Toronto, Canada. The 14th and 43rd floor serve as the mechanical rooms, which service the lower and upper half of the building. For the purpose of this analysis, the upper zone refers to floors 29 to 55, while the lower zone refers to floors 2 to 28. HVAC zoning is split into east and west for both the perimeter and interior core zones. Each zone is further split into a north and south subzone. The plaza is an expectation to rule as it only has the west and east zone designation.

The key parameters to track were derived from Yoon et al. (2015), who identified the critical data points required to create a physics-based model for stack effect. Based on their work, the parameters for this research were distilled into the following categories: weather data, HVAC trends, and interior temperature. The variables captured from the sensor network are listed in Table 1. These variables were chosen as they are critical for creating a physics-based model, and did not require labour extensive testing. Building envelope air tightness, and leakage rates were also key parameters to create a multi-zone model but they were omitted as it was infeasible to measure these variables due to Covid-19 restrictions. Weather data was recorded from Tomorrow.io, a weather API (Tomorrow.io, 2020), while HVAC trends were pulled from Metasys, the building automation system (BAS) program. Pressure differential sensors data was not calibrated at this time due to delays related to Covid-19, but are planned to be implemented in future iterations of the this research. Data was collected from November 2020 to April 2021 to capture any seasonal variations from the heating season to the beginnings of the cooling season.

Table 1 Data Collected for G2 Model Development

Category	Parameter	Unit	Sensor/Support Program
Weather Data	Exterior Temperature	°C	Tomorrow.io
	Relative Humidity	%	
	Wind Speed	km/h	
	Wind Gust	km/h	
	Wind Direction	°	
HVAC Trend	Supply Air Temperature	°C	Metasys
	Return Air Temperature	°C	
	Fan Flow Rate	m^3/s	
Interior Temperature	Average Interior Temperature (based on VAV readings)	°C	

3.2 Data Visualization and Pre-processing

Data visualization was used to develop an understanding of the data behavior and identify any potential outliers, which was done in conjunction with data pre-processing. Data pre-processing is the stage right before model development, where the data is cleaned and resampled in order to create suitable dataset for machine learning. Data pre-processing was achieved through python using the pandas library (McKinney, 2011) for data manipulation. Outliers were removed based on the observations from the visualized data, and resampled to one minute intervals to keep the time series uniform for each data point. Missing values were linearly interpolated, and the resultant dataset was outputted onto an excel file.

In previous work (Chang, et al., 2021), return air temperature (RAT) was used as the instrumental variable to detect stack effect-related thermal comfort issues. For this new work, this approach was revised to consider the difference between average floor temperature and the duct-mounted return air temperature as a proxy to identify air infiltration caused by stack effect. A range of temperature thresholds were considered for the lower and upper RAT, which is explained in the results. For the modeling results, thresholds were set recognizing that infiltration has a more significant effect on average temperature than exfiltration, resulting in a temperature differential above 1°C indicating air exfiltration due to stack effect in the upper half of the building, and above 1°C to indicate air infiltration due to stack effect in the lower half of the building. These temperature thresholds were chosen based on expert judgment based on envelope tightness, coupled with data observations to minimize noise generated from other factors such as occupancy. The label classification is presented in Table 2.

Table 2 Label Classification for RAT as Proxy for Stack Effect

	Upper RAT – Upper Avg Floor Temp Difference < 1°C	**Upper RAT – Upper Avg Floor Temp Difference >= 1°C**
Lower RAT – Lower Avg Floor Temp Difference < 1°C	Green - No air infiltration due to stack effect	Green– Air exfiltration at upper half but no air infiltration on lower half
Lower RAT – Lower Avg Floor Temp Difference >= 1°C	Green –air infiltration but not exclusive to stack effect	Red –Air infiltration and exfiltration due to stack effect

3.3 Model Training

The data-driven model classifies the instances in the dataset to calculate potential for stack effect related issues in the case study building. The labels created based on the temperature difference between the average floor temperature and RAT served as the output of the model, which indicated potential excess air infiltration into the building due to stack effect. Weather data, and the HVAC trend (except for RAT) listed in Table 1 served as the inputs of the model. The inputs were chosen since they have a physics-based relationship which will aid the model in learning process. Because of the dearth of previous work in this field, a 'brute force' approach was used for algorithm selection, whereby each algorithm was tested and ranked based on accuracy, recall, precision, and observations made from the confusion matrices.

The dataset was split into 60% for training, 20% for validation, and 20% for testing. The models were trained and tested using data that spanned from March 2021, where the results of the G2 model were compared to the G1 model. To respect the temporal aspect of the data, the model is trained on data in the past, and predicts using a test set from the future. The model is trained and tuned using the training, and validation sets, with the final 20% reserved for final testing. Each algorithm was trained and tested on the same dataset to avoid any biases in the results. Model training was implemented using python through the scikit-learn package that simplifies the workflow (Pedregosa, et al., 2011). As identified in the literature review, the following algorithms are best suited for modelling stack effect: LDA (Li, et al., 2016), KNN, RF, SVM (Shohet, et al., 2020), and MLP (Capozzoli, et al., 2015).

3.4 Model Validation and Recalibration

The models were validated using fundamental classification metrics (Géron, 2017) of recall/sensitivity, f1 score, precision, and accuracy. Accuracy calculates the percentage of correct predictions from the model, while recall, f1_score, and precision are used to measure the variance and biases in the results. They provide insight on the overall generalization of the model. No information rate (NIR), also known as null accuracy, is the accuracy a model would achieve if the largest class label was always picked. This was used a baseline to indicate if the model showed any statistical significance, where if the model had an accuracy lower than the NIR it would imply the model did not generate statistical significant results.

Based on the metrics, the hyperparameters of each model was recalibrated to optimize the cost function, and consequently the model accuracy. The grid search technique was used which tests every possible set of hyperparameters for each algorithm to determine the best accuracy it can achieve with the training dataset. The downside to the grid search technique is the high computational cost as many iterations of each model is computed, but it is the most thorough method. Feature selection was done through eliminating features that contained a collinearity of 0.97 or more, and data normalization were done to improve model performance. After every algorithm has been optimized, the best model is chosen based on the accuracy, statistical significance, and the generalization capabilities.

Table 3 List of Tested Hyperparameters

Algorithm	Hyperparameter	Range	Optimal for G1	Optimal for G2
LDA	Solver	{svd, lsqr}	Svd	svd
KNN	n_neighbors	{3,5,7}	3	7
RF	n_estimators	{25,50,100}	100	25
	max_depth	{1,2,3}	3	3
	min_samples_split	{2,5,7}	7	5
	min_samples_leaf	{1,2,5}	1	2
MLP	Solver	{lbfgs, sgd, adam}	sgd	adam
SVM	Kernel	{linear, rbf, poly, sigmoid}	poly	rbf
	C	{1,10,50}	1	1
	Gamma	{scale, auto}	auto	auto
	Degree (for poly)	{1,2,5}	2	1

3.5 Digital Twin

The DT (Figure 3) is currently under development and is based on an FM-BIM created in Autodesk Revit using Lean-Agile approach (McArthur & Bortoluzzi, 2018) to support the stack effect use case. As such, this model included the envelope components (walls, slabs, and openings), thermal zones, and all HVAC air-side equipment.

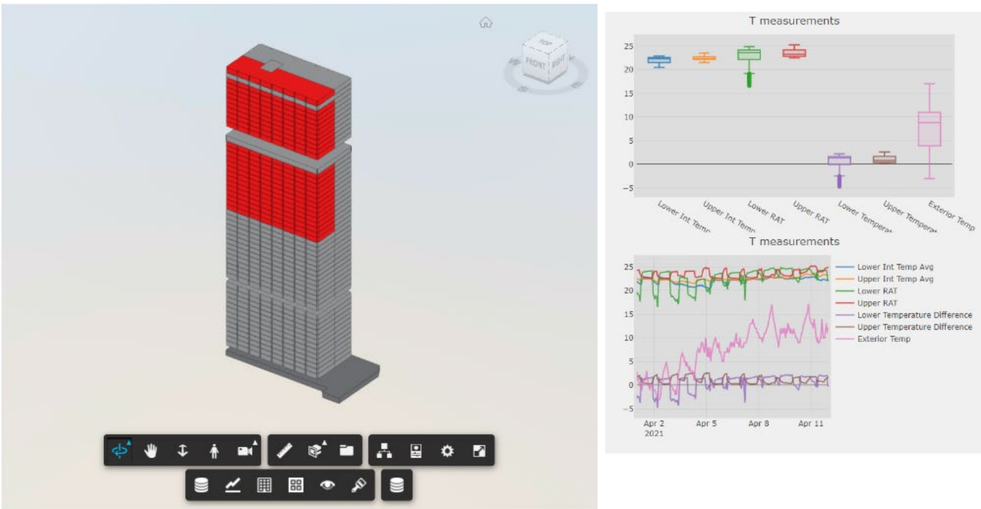

Figure 3 Digital Twin of Case Study Building

Because of the limited computational power within BIM, the DT pulls the BIM software, the model itself is a consumer of the data, with all data analytics undertaken using python and mapped into the DT using approach developed by Quinn et al (Quinn, et al., 2019), which transforms data from the data lake into nested tables, grouped by time, element ID, and point ID, respectively. This approach allows the desired time to be selected in a Dynamo interface for the model and all points to be updated with the latest data, including the predicted algorithm results and subsequent recommendations.

4 Results

Data was extracted from November 2020 to April 2021 to develop an understanding of the trends in the case study buildings. The results of the analysis are presented are organized as follows: (1) Data Visualization; (2) Model Development.

4.1 Data Visualization

Return air temperature from February 2021 and April 2021 were plotted against the outdoor air temperature to visualize the trends, shown in Figure 4. As illustrated in Figure 4 for February 2021, the upper RAT remained relatively consistent throughout the month but the lower RAT saw significant drops in temperature during the nights and weekends. The lower RAT had a higher correlation with the exterior temperature than the upper RAT, which implied the presence of significant air infiltration on the lower levels. The upper RAT did not experience this behavior which indicated a presence of a constant heat source in the upper levels, providing evidence of stack effect based on RAT as a proxy. This was further validated by the thermal circuit model presented in the previous iteration of this research (Chang, et al., 2021). Figure 4 for the month of April 2021, visualizes the RAT and OAT as the weather gradually transitions to the cooling season. The correlation between RAT and exterior temperature followed the same trend as in February, with the lower RAT correlating better. The trends in the lower RAT begin to remain more consistent as the outdoor air temperature increases. This observation reinforces the possibility of stack effect affecting the RAT as it expected the lower RAT will vary less during warmer weather due to smaller pressure differentials, and consequently less stack effect.

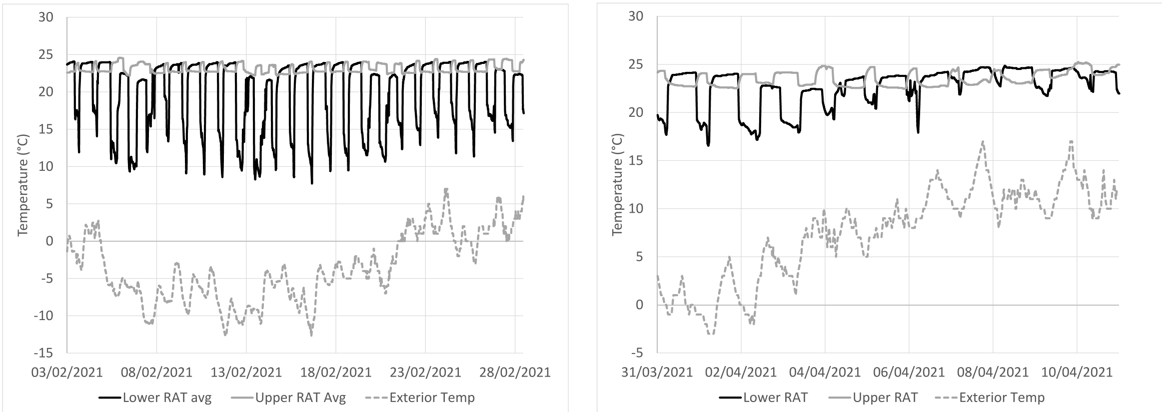

Figure 4 Average RAT vs OAT – February 2021 (Left) and April 2021 (Right)

The temperature difference between the RAT and average floor temperature of the upper and lower zones are plotted on Figure 5. As illustrated in Figure 5 for February 2021, the lower half of the building saw significantly larger differences than the upper half, where a 1°C difference was chosen for the lower RAT threshold for classification as explained in section 3.2, to account for noise in the data. The temperature differentials during April 2021 showed good agreement with the observations from Figure 4 of the same month, as the differentials reduced in magnitude when the outdoor air temperature was greater.

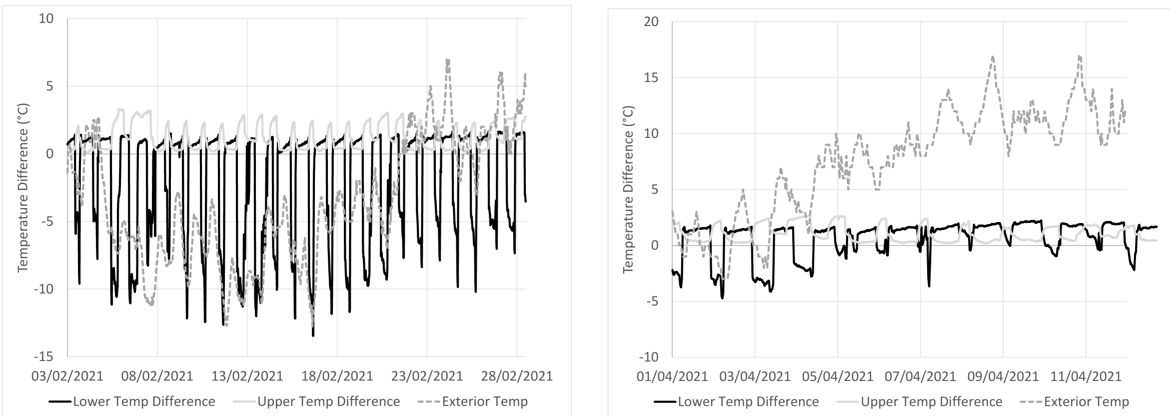

Figure 5 RAT - Floor Temp Difference – February 2021 (Left) and April 2021 (Right)

4.2 Model Development

The G1 and G2 were tested to compare the classification of the two iterations of the data-driven model. The updated G1 model used the same input feature set as the G2 model {supply fan temperature, fan flow rate, wind gust, wind speed, exterior temperature, relative humidity, hour, day}. Both were trained and tested with the same data sets. As shown in Tables 4 & 5, all algorithms were able to output statistically significant results with SVM being the best performer with an accuracy of 92.2%. This finding differed from the winter-only G1 results (Chang, et al., 2021), most likely due to the greater difference between the training and testing datasets as the weather transitioned to the shoulder season. The results from the G2 model are presented in Table 5. MLP was deemed the best G2 model overall as it had high accuracy and f1_score which indicated good generalization capabilities. Conversely, LDA was the worst performer and did not provide statistically significant results. For the best performers of both models (SVM for G1 and MLP for G2), the green label was more commonly misclassified indicating a false positive for stack effect related issues. Note that there was significant imbalance in the initial dataset due to the significant number of shoulder season data points, and thus this was resampled to balance the classes in order to improve the recall of all algorithms.

Table 4 Generation 1 (G1) Model Results

Algorithm	LDA	KNN	RF	MLP	SVM
Accuracy	0.586	0.865	0.912	0.908	**0.922**
Precision	0. 607	0.868	0.902	0.913	**0.912**
Recall	0.613	0.839	0.925	0.888	**0.933**
f1_score	0.584	0.850	0.909	0.898	**0.919**
NIR	0.632	0.632	0.632	0.632	**0.632**

Table 5 Generation 2 (G2) Model Results

Algorithm	LDA	KNN	RF	MLP	SVM
Accuracy	0.565	0.934	0.898	**0.955**	0.928
Precision	0.592	0.919	0.884	**0.945**	0.912
Recall	0.600	0.947	0.919	**0.959**	0.941
f1_score	0.562	0.929	0.892	**0.951**	0.922
NIR	0.659	0.659	0.659	**0.659**	0.659

To analyze the performance, the labels were plotted based on instances where the RAT indicated stack effect related issues which was the red label for both the G1 and G2 model. The other labels were classified as no stack effect related issues for the purpose of this analysis. As shown in Figure 6, using the 5°C lower RAT threshold that was initially proposed for the G2 model resulted in both models having almost identical classifications for instances where stack effect is not problematic, but differed when it came to raising the stack on stack effect issues. The confusion matrix showed the G1 model classified 16% more of the instances red than compared to the G2 model. As the temperature threshold was lowered, the classifications from the G2 model gradually grew more similar to the G1 until it reached a 1°C threshold at which the classifications were almost identical. This is shown in Figure 6, as the classifications resulted in less than a 7% difference. Based on these results the G1 model may be too sensitive to the noise in the dataset resulting in a higher percentage predictions classifying the instance as potential for stack effect related issues. As the G2 model uses a more robust approach, calculating the difference between the RAT and average floor temperature, and using an appropriate temperature buffer some of this noise in the data may be mitigated for better model performance. Reviewing the time series plots of the G1 and G2 labels (where 1 indicated stack effect issues and 0 meant no stack effect issues) showed that the G1 model frequently classified instances as an issue more frequently, especially during early March. It should also be noted that the majority of discrepancies occur during the operating hours on the weekday, which is further indication that the G1 model is sensitive to the noise generated from building occupancy.

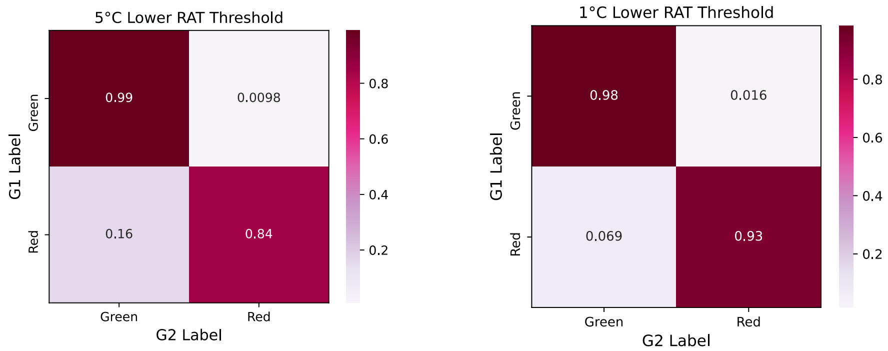

Figure 6 G1-G2 Classification Matrix- 5°C Lower RAT (Left) and 1°C (Right) Lower RAT Thresholds

5 Discussion & Conclusion

Based on the results, the G2 model is a more robust framework for developing a data-driven model of stack effect based on RAT as a proxy when compared to the G1 model approach from the previous study. As shown in the confusion matrix, the G1 model is more prone to predicting instances where stack effect issues are arising. By using a 1°C temperature buffer to account for any noise in the dataset the G2 model reduced the number predictions resulting in stack effect related issues. Both the G1 and G2 models showed good results overall, with SVM being the best for the G1 and MLP for the G2. The high f1_score, precision, and recall also indicated that the algorithms showed good generalization of the model. The fact that KNN showed improved performance for the G2 model indicated greater separation between the classes, further reinforcing the hypothesis that the G2 model is a stronger approach moving forward due to the robustness of the classification method and the strong modelling results.

Due to delays related to Covid-19, the installed pressure differential sensors have yet to be calibrated in time for this publication but is planned for use in future work. Using the sensors, the G1 and G2 models will be validated through the definitive approach of pressure differentials to gauge the severity of stack effect in the case study building. Another limitation was the lack of cooling season data available to test and train on, which will have a vastly different behavior when compared to the heating and shoulder seasons. By implementing more data into this process the generalization capabilities of these models can be improved in the future.

This research expanded upon the possibility of using RAT as a proxy for stack effect by evaluating it against the average floor temperature to give indication of potential air infiltration. MLP has proven to be the best algorithm during this round of testing, and could be potentially the best due to the ability to solve non-linear problems. As the framework for the digital twin is in place, future work will focus on validating the RAT as a proxy model with the differential pressure data. This will lead into the final phase of this research in which the classification models will feed into the development of a recommender algorithm to create a stack effect mitigation tool for facilities management.

6 Acknowledgements

This research is funded by Natural Sciences and Engineering Research Council (NSERC) [EGP 543443-2019] and HH Angus & Associates Ltd. The student author was also funded by the NSERC [CREATE 510284-2018] and additional funding from the Toronto Hydro Corporation Student Award.

7 References

Axley, J., 2007. Multizone Airflow Modeling in Buildings: History and Theory. *HVAC&R Research,* 13(6), pp. 907-928.

Capozzoli, A., Lauro, F. & Khan, I., 2015. Fault detection analysis using data mining techniques for a cluster of smart office buildings. *Expert Systems with Applications,* 42(9), pp. 4324-4338.

Chang, G., El Mokhtari, K., Richman, R. & J.J., M., 2021. *Evaluating Stack Effect Impact of Thermal Comfort in High Rise Office Towers.* Bruges, Building Simulation 2021.
Chen, Q., 2009. Ventilation performance prediction for buildings: A method overview and recent applications. *Building and Environment,* 44(4), pp. 848-858.
Géron, A., 2017. *Hands-On Machine Learning with Scikit-Learn Concepts, Tools, and Techniques to Build Intelligent Systems.* Sebastopol, California : O'Reilly Media, Inc..
Li, D., Hu, G. & Spanos, C. J., 2016. A data-driven strategy for detection and diagnosis of building chiller faults using linear discriminant analysis. *Energy and Buildings,* Volume 128, pp. 519-529.
Lim, H. et al., 2020. Interaction analysis of countermeasures for the stack effect in a high-rise office building. *Building and Environment,* Volume 168, p. 106530.
Lim, T., Cho, J. & Kim, B. S., 2011. Predictions and measurements of the stack effect on indoor airborne virus transmission in a high-rise hospital building. *Building and Environment,* 46(12), pp. 2413-2424.
Lovatt, J. E. & Wilson, A. G., 1994. Stack effect in tall buildings. *ASHRAE Transactions,* Volume 100(2), pp. 420-431.
Madeti, S. & Singh, S., 2018. Modeling of PV system based on experimental data for fault detection using kNN method. *Solar Energy,* Volume 173, pp. 139-151.
McArthur, J. & Bortoluzzi, B., 2018. Lean-Agile FM-BIM: a demonstrated approach. *Facilities,* 36(13/14), pp. 676-695.
McKinney, W., 2011. pandas: a Foundational Python Library for Data Analysis and Statistics. *Python High Performance Science Computer,* 14(9).
Molina-Solana, M. et al., 2017. Data science for building energy management: A review. *Renewable and Sustainable Energy Reviews,* Volume 70, pp. 598-609.
Pedregosa, F. et al., 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research,* Volume 12, pp. 2825-2830.
Quinn, C., Shabestari, A., Litoiu, M. & McArthur, J., 2019. *Building Automation System Data Integration with BIM: Data Structure and Supporting Case Study.* Newcastle, scix.net.
Shohet, R., Kandil, M. S., Wang, Y. & McArthur, J. J., 2020. Fault detection for non-condensing boilers using simulated building automation system sensor data. *Advanced Engineering Informatics,* Volume 46, pp. 1-16.
Tamblyn, R. T., 1991. Coping with air pressure problems in tall buildings. *ASHRAE Transactions,* Volume 97, pp. 824-827.
Tomorrow.io, 2020. *Tomorrow.io API.* [Online] Available at: https://www.tomorrow.io/weather-api/
Wilson, A. & Tamura, G., 1968. Stack effect in buildings. *Canadian Building Digest,* Volume 104.
Wong, N. H. & Heryanto, H., 2004. The study of active stack effect to enhance natural ventilation using wind tunnel and computational fluid dynamics (CFD) simulations. *Energy and Buildings,* 36(7), pp. 668-678.
Yoon, S., Seo, J., Cho, W. & Song, D., 2015. A calibration method for whole-building airflow simulation in high-rise residential buildings. *Building and Environment,* Volume 85, pp. 253-262.
Yoon, S., Song, D., Kim, J. & Lim, H., 2019. Stack-driven infiltration and heating load differences by floor in high-rise residential buildings. *Building and Environment,* Volume 157, pp. 366-379.
Yu, J.-Y., Cho, D.-W., Yu, K.-h. & Jung, H.-K., 2004. *Evaluation of stack effect according to the shape and the window area ratios of lobby in highrise buildings.* Seoul, s.n.
Yu, J.-Y., Song, K.-D. & Cho, D.-W., 2017. Resolving Stack Effect Problems in a High-Rise Office Building by Mechanical Pressurization. *Sustainability,* 9(10), p. 1731.
Zhang, J. Y. et al., 2006. A Comparison of Simulation and Experiment on Stack Effect in Long Vertical Shaft. *Journal of Fire Sciences,* 24(2), pp. 121-135.