BUILD SIMUL
https://doi.org/10.1007/s12273-020-0637-y

# Extracting typical occupancy schedules from social media (TOSSM) and its integration with building energy modeling

**Xing Lu, Fan Feng, Zhihong Pang, Tao Yang, Zheng O'Neill (✉)**

*Department of Mechanical Engineering, Texas A&M University, College Station, TX 77843, USA*

Research Article

## Abstract

Building occupancy, one of the most important consequences of occupant behaviors, is a driving influencer for building energy consumption and has been receiving increasing attention in the building energy modeling community. With the vast development of information technologies in the era of the internet-of-things, occupant sensing and data acquisition are not limited to a single node or traditional approaches. The prevalence of social networks provides a myriad of publically available social media data that might contain occupancy information in the space for a given time. In this paper, we explore two approaches to extract the typical occupancy schedules for the input to the building energy simulation based on the data from social networks. The first approach uses text classification algorithms to identify whether people are present in the space where they are posting on social media. On top of that, the typical building occupancy schedules are extracted with assumed people counting rules. The second approach utilizes the processed Global Positioning System (GPS) tracking data provided by social networking service companies such as Facebook and Google Maps. Web scraping techniques are used to obtain and post-process the raw data to extract the typical building occupancy schedules. The results show that the extracted building occupancy schedules from different data sources (Twitter, Facebook, and Google Maps) share a similar trend but are slightly distinct from each other and hence may require further validation and corrections. To further demonstrate the application of the extracted Typical Occupancy Schedules from Social Media (TOSSM), data-driven models for predicting hourly energy usage prediction of a university museum are developed with the integration of TOSSM. The results indicate that the incorporation of TOSSM could improve the hourly energy usage prediction accuracy to a small extent regarding the four adopted evaluation metrics for this museum building.

## Keywords

occupancy schedule,
social media,
building energy modeling,
data-driven models

## Article History

Received: 30 December 2019
Revised: 19 February 2020
Accepted: 24 March 2020



Building Thermal, Lighting, and Acoustics Modeling

## 1 Introduction

Occupant behaviors in buildings have become a hot topic with building systems getting more sophisticated and people spending significant time in buildings (Abergel et al. 2017). Occupants and their behaviors are known as a driving factor of the building energy consumption. They have a direct impact on the accuracy of building energy modeling (Yu et al. 2011; Muroni et al. 2019), operation and control of intelligent building systems (Naylor et al. 2018; Park et al. 2019), as well as the design of the future building system (Samuelson et al. 2016). Therefore, knowing the presence, number, variation, and comfort requirements of occupants in buildings is a key component of the occupant-oriented research (Dong et al. 2019).

A large number of cases studies have been conducted in the past decades to investigate both commercially and computationally achievable ways to extract the occupancy for the building energy applications. Among these, sensor technology is a prevalent way to obtain occupancy information in both academia and industry, mostly due to its easy implementation and high feasibility. The most commonly used sensing technique for the occupancy in buildings is a passive infrared (PIR) sensor (Agarwal et al. 2010), which falls into the category of movement-based sensors, including ultrasonic doppler sensors, sound sensors, etc. (Dong et al.

E-mail: ZONeill@tamu.edu

Nomenclature

| | | | |
|---|---|---|---|
| AHU | air handling unit | Occ | occupancy |
| API | application program interfaces | PCC | Pearson correlation coefficient |
| CHW | chilled water | PIR | passive infrared |
| CV(RMSE) | coefficient of variation of root-mean squared error | $R^2$ | *R*-squared |
| | | RGB | red, green, blue |
| FP | false positives | RF | Random Forest |
| FN | false negatives | SVM | support vector machine |
| GPS | Global Positioning System | TFIDF | term frequency-inverse document frequency |
| HW | hot water | TOSSM | typical occupancy schedules from social media |
| IoT | Internet of things | TP | true positives |
| IP | Internet Protocol | TN | true negatives |
| MAC | media access control | URL | uniform resource locator |
| MAE | mean absolute error | XGB | XGBoost |
| NMBE | normalized mean bias error | | |

2019). These sensors can generate an output value of one or zero in each time step, which represents the binary data, "occupied" and "unoccupied" status, of the space, respectively. Despite their broad applications, the inherent issues with such binary sensors are that they can only provide the occupancy presence information instead of people counting. Hence, they are not likely to be used in the load-oriented control cases for modern intelligent building controls (Pang et al. 2020). To address this limitation, some other occupancy detection technologies, such as vision-based technologies (e.g., RGB camera, infrared thermal camera) (Jazizadeh and Jung 2018) and environment-based technologies ($CO_2$ sensor, etc.) (Jin et al. 2018) are introduced. These approaches, sometimes coupled with the movement-based sensors, can assist in detecting the number of people in the room (Jung and Jazizadeh 2019). Regardless of this fact, the occupant detection approaches still have privacy concerns (image-based) and delayed response issues (ambient-based). Besides, initial costs are always a barrier for large-scale adoption of both presence and counting sensing system.

Considering the initial investment, some studies proposed to use the existing sub-metering and infrastructure systems (like applicants and communication systems) in the buildings to extract the occupancy information. For example, Newsham et al. (2017) conducted a field study to test the accuracy of various IoT data stream for detecting the occupancy in the office. They discovered that a combination of keyboard/mouse activity and pixel change in a webcam image could provide a better occupancy detection than incumbent commercial sensors, such as the PIR sensor. Another example is that Christensen et al. (2014) extracted the occupancy schedules of two buildings based on the existing IT infrastructure (i.e., the Wi-Fi network). In detail, they monitored and mapped the IP and MAC addresses of Wi-Fi access points and routers to the occupants of each space in the building, and therefore the occupancy schedules are created. The existing infrastructure-based occupancy extraction methods have the advantages of no additional costs in terms of hardware and installation. However, they are only suitable for those buildings in which the infrastructure is well-functioned and available. Another communication approach that does not depend on the building infrastructure system is the Global Positioning System (GPS). An in-depth analysis of the massive location data generated by the mobile service users could also be used to create the occupancy schedules at the building level. Pang et al. (2018) monitored the occupancy variation of an office building in Shanghai, China using the location data shared by the smartphone users when they use online services such as food delivery, carpooling, navigation, etc. Based on the monitored results, an occupancy schedule was generated to facilitate a building energy model calibration. Besides, Gu et al. (2018) extracted the typical occupancy schedules for various building types using the same data source. Despite its merits of no hardware and installation costs, this method suffers from the issue of privacy violation, because these raw data are all collected from users' private information.

The merits and demerits of the aforementioned occupancy detection methods are summarized in Table 1. Although these studies show promising potential to extract the building occupancy information, their drawbacks are also non-negligible, e.g., the sensor error, high cost, scalability, and privacy issues, which hinder a broad implementation of occupancy sensing in buildings. Therefore, alternative data sources for building occupant behavior extraction should be considered and explored.

**Table 1** The merits and demerits of the normal occupancy extraction methods

| Methods | Typical sensors | Occupancy information | Merits | Demerits |
|---|---|---|---|---|
| Movement-based technology | PIR sensor<br>Ultrasonic sensor<br>Sound sensor | Presence | Easy installation and low costs | Only the binary presence information is available<br>Additional costs<br>Intrusion |
| Vision-based technology | RGB camera<br>Infrared thermal camera | Presence/ Counting | The occupant number is available | Privacy issues<br>High costs<br>Intrusion |
| Environment-based technology | $CO_2$ sensor<br>Temperature/Humidity sensor | Presence/ Counting | The occupant number is indirectly obtained<br>Non-intrusive | Delayed response |
| IoT-based technology | Keyboard/mouse<br>Pixel webcam | Presence/ Counting | No additional costs for the hardware and installation.<br>Non-intrusive | Privacy issue<br>Scalability issue: complete and well-functioned building infrastructure is needed |
| Communication-based technology | Wi-Fi network<br>Mobile-GPS | Presence/ Counting | No additional costs for the hardware and installation<br>Suitable for all buildings<br>Not intrusive | Privacy issue<br>Scalability issue: mobile infrastructure is needed |

To fill this gap, this paper explored another occupancy detection approach, which takes the advantage of the social media data posted by the users voluntarily and publicly. The prevalence of social networks provides a myriad of publicly available social media data that contains occupancy information in space and in time (Lu et al. 2019). However, only a few existing studies were targeted at using this data source to estimate the building occupancy. The Population Density Tables (PDT) project by Oak Ridge National Laboratory estimated the ranges for an average day and night population density for over 50 building types using the Bayesian learning model with different open source data (Stewart et al. 2016). Stewart et al. (2017) proposed a social network unit occupancy model to extract the social media-based occupancy curve for a museum during its operating hours. Sims et al. (2017) applied social media data to conduct a high-resolution mapping of a special event population. Twitter posts and Facebook check-ins were calculated for the Game Day at the University of Tennessee Knoxville. Population distributions for game hours and nongame hours of the game day were modeled using social media data. It is noted that it used a linear relationship to describe the event population with social media activity. Bentz et al. (2019) designed a thermostat in which the setpoint could be adjusted based on the expected occupancy and the social media activity. These studies indicate the feasibility of extracting the building occupancy information from social networks. However, none of these studies moved further to explore its integration with the building energy modeling, and research on its influence on the modeling accuracy.

In this paper, we propose two different non-intrusive, cost-free, low-privacy-sensitive approaches, based on the data from social networks for extracting the typical occupancy schedules. These schedules then act as inputs for the building energy simulation. In the meanwhile, to demonstrate the application of the extracted building occupancy schedules and evaluate their values, data-driven building energy models for a university museum are constructed to see whether additional feature regarding the occupancy at the building level will facilitate the improvement of the prediction accuracy and the fidelity of the building energy model.

The paper is organized, as illustrated in Figure 1. Section 2 described two proposed methods (i.e., approach 1 of text classification through Tweets and approach 2 of web-scraping from Facebook/Google Maps) to extract typical occupancy schedules from social media with a case study using a public museum building. Section 3 demonstrates the integration of the extracted typical occupancy schedules through approach 2 into a data-driven building energy prediction model of a university museum. Section 4 presents the conclusions, limitations, and future work.

## 2 Extraction of typical occupancy schedules from social media (TOSSM): case study 1 for a public museum building

### 2.1 Overview

In this section, two different approaches for extracting the typical building occupancy schedules at the building level

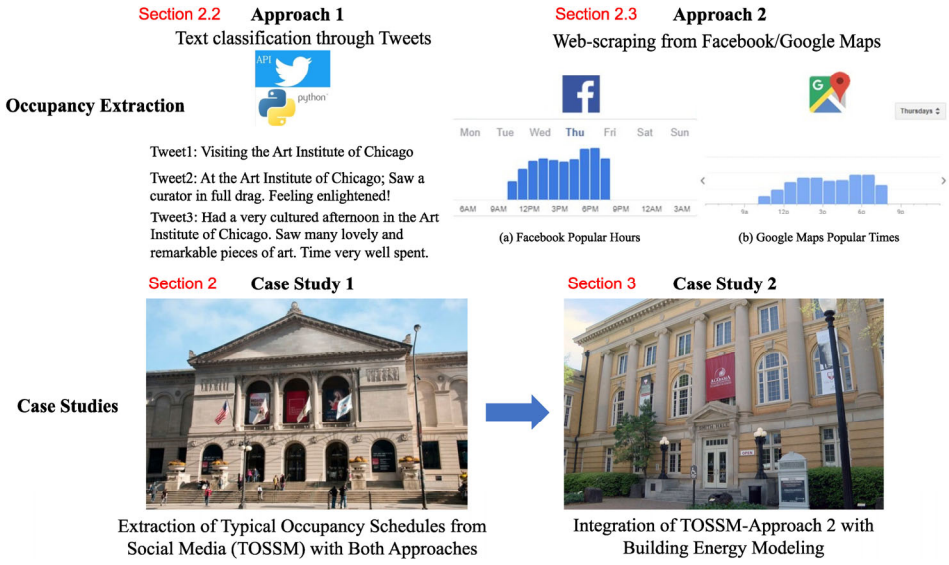



**Fig. 1** Schematics of paper organization and sections relationship

are explored based on the data from social networks. The first approach is to use text classification algorithms to identify whether people are present in space where they are posting on social networks (e.g., Tweeter). To achieve this, word embedding and machine learning algorithms for classification are used. On top of that, the typical occupancy schedules could be extracted by assuming certain people counting rules. The second approach is to utilize the processed GPS tracking data provided by social networking service companies such as Facebook and Google Maps. Web scraping techniques are used in this process to obtain the raw data and extract the typical occupancy schedules at the building level.

The Art Institute of Chicago, a public museum, is selected as a case study building. The Art Institute of Chicago, founded in 1879 and located in Chicago's Grant Park, is one of the oldest and largest art museums in the United States. It opens daily from 10:30 to 17:00 except on Thursdays until 20:00. The reason why we select a public museum building in this study is that there is a higher chance of people creating posts about their visits to such a tourist attraction. Besides, both Facebook and Google Maps provide popular time information on their websites for this type of building. Therefore, more datasets could be obtained to facilitate the comparison of the two approaches.

## 2.2 Text classification of implicitly geo-tagged posts from Tweets

Utilizing public application program interfaces (APIs) provided by the social media services, it is possible to attain the geographic information through either geo-tagged posts from Twitter or Facebook check-in messages, which is depicted in Figures 2 (a) and (b). These datasets explicitly indicate the occupant presence and could be used to estimate the occupancy. However, it is well known that most social media users probably are not willing to disclose their location information. Although the datasets from the explicitly geo-tagged posts could be insufficient to represent the occupancy information, the implicitly geo-tagged posts could


**Fig. 2** Explicit geo-tagged posts: (a) geo-tagged posts, (b) check-in posts; implicit geo-tagged posts: (c) example 1, (d) example 2

be a workaround as another social media data source for occupancy sensing. These geo-tagged posts are those that could be inferred for the human occupancy, but the user does not add his/her location to the posts. Figures 2 (c) and (d) show two examples of the implicitly geo-tagged posts. We could infer from the Tweet textual semantics that the user is currently in the building, that is, the Art Institute of Chicago. However, there are some cases that the users mentioned the detailed location in the post, but they are apparently not present at a certain location. For example, in the following posts: "I've always wanted to go to the Art Institute of Chicago. # bucketlist"; "Hotels near the Art Institute of Chicago https://www.govisitchicago.com/top-hotels-near-art-institute-chicago/."

Text classification and semantic analysis could be utilized to help us identify the right implicitly geo-tagged posts, which contain the occupancy information. Text classification problems have been widely used and addressed in many real applications, such as information retrieval, sentiment analysis, recommender systems, etc. (Kowsari et al. 2019). To increase the volume of the social media datasets in the building occupancy applications, we present a methodology to detect the implicitly geo-tagged posts from the social media that hold valuable occupancy information to sense the occupancy in buildings at the building level.

This approach involves four essential procedures: data collection and pre-processing, feature generation, classifier formulation, and result evaluation, as illustrated in Figure 3. Each of these four procedures will be described with details in the following subsections.

### *2.2.1 Data collection and preprocessing*

One way of collecting the data is through the official APIs of social networking service providers. The U.S. social media giants Twitter, Facebook, and Reddit all have their proprietary APIs. However, this approach has some limitations for free and standard users. Take the Twitter Standard Search API as an example; the free standard tier allows the return of at maximum 100 relevant Tweets in the seven days. The data fidelity is incomplete compared to the paid categories. The paid access could allow the developer access to the full-fidelity data from as early as 2006, along with direct account management support, and dedicated technical support to help on an integration strategy. Another way of collecting data is through web-scraping. As aforementioned, official APIs have the limitation of time constraints; therefore, we cannot get tweets older than a week. However, web-scraping tools such as GetOldTweets (Henrique 2019) could provide us with history posts. The basic underlying principle is summarized as follows. When we enter a Twitter page, a scroll loader would automatically start. If we scroll down, we will get more and more tweets with the scroll loader. The GetOldTweets tool exactly mimics this process. In this way, we could take the best advantage of Twitter Search on browsers and deeply search the oldest tweets.

All data needs to be cleaned before the feature extraction and being fed to the classifier, which can help to reduce the noise in text data. Most text data from social media contain many unnecessary words such as stop words, misspelling, slang, etc. Many text-processing techniques are suggested, such as tokenization, stop word elimination, case lowering, slang and abbreviation paraphrase, spelling correction, stemming, lemmatization, etc.

We collected the history posts between April and May 2019 using the Twitter official API approach and its counterpart, the GetOldTweets approach. We searched the relevant Tweets using the keywords "art," "institute," and "Chicago." We compared the data from the two data sources and found that the data size is smaller for the second approach. However, we also found that the data from GetOldTweets neglected the retweet posts and the posts that are existing in history. The others are the same for these two methods. Considering that the retweet posts and the existing posts in the history typically do not indicate the presence of the people, it would be suggested to use the

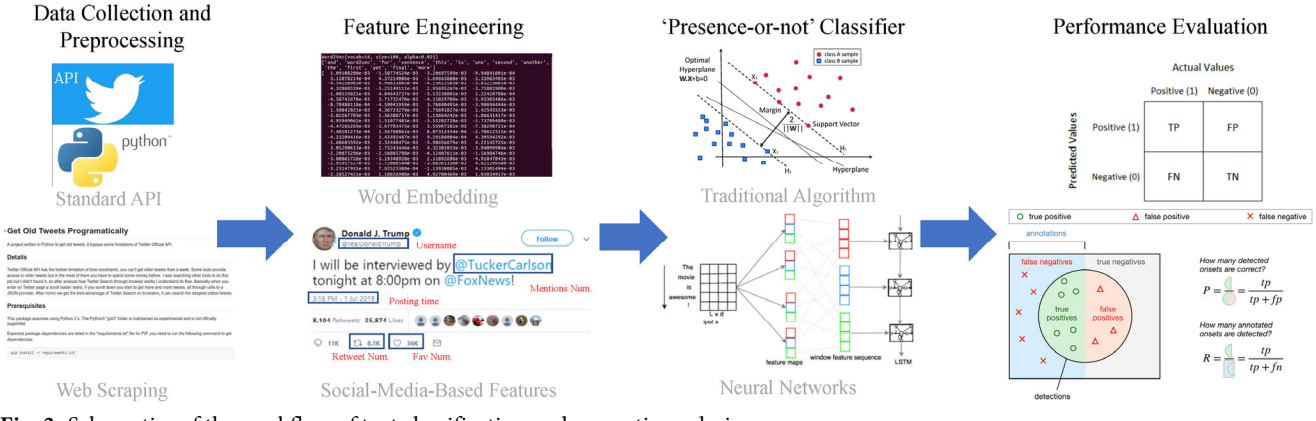



**Fig. 3** Schematics of the workflow of text classification and semantic analysis

GetOldTweets approach since it is free of charge and also have a similar amount of data compared with the official API approach.

On top of that, we collected all the available history posts from December 2016 to June 2019 (approximately 30,000 in total) using the GetOldTweets approach and manually labeled the latest 3,000 history posts, which indicated whether the user was present or not. It is found that the positive (people-presence) data only occupies ~15% of all the labeled data. To balance the proportion of the true positives and true negatives, we use all the true positives. The total number of the training and validation datasets is 1,000. For the data pre-processing, we lower the case of the posts, conduct the tokenization, and then remove the stop words.

#### *2.2.2 Feature engineering*

In this step, the raw text data will be transformed and processed into the feature vectors. Different categories of features will be combined to help improve the accuracy of the classifier, such as weighted words, word embedding, as well as social media-based features. The first two methods are typical feature extraction methods with the text data while the third method is based on the characteristics of the social media posts.

For the weighted words, the Bag-of-Words (BoW) model (Wallach 2006) and Term Frequency-Inverse Document Frequency (TF-IDF) (Wu et al. 2008; Wikipedia Contributors 2020) are two commonly used approaches. The BoW is represented as the bag of its known words where the occurrence of each word is used as a feature. TF-IDF is a statistical measure that weighs down the frequent words and scales up the rare ones to reflect the word importance in a corpus. Both methods are easy for the implementation. However, they only produce the counting and importance of the single word and do not capture the position and the meaning in the text. Word embedding models could capture the semantics of the word, and each word will be mapped to an N dimension vector of real numbers. A word embedding is a form of representing words using a dense vector representation. Word2Vec (Mikolov et al. 2013), GloVe (Pennington et al. 2014), and FastText (Bojanowski et al. 2017) are the three most common pre-trained models to keep the syntactic and semantic information of each sentence. Apart from the pre-trained word embedding, we could also learn the word embedding layer as a part of fitting a machine learning model. Social media-based features are statistical features based on the characteristics of the social media posts such as the presence of URLs, the presence of hashtags, hashtag count, favorite count, repost count, etc. Different combinations of these features will be fed into a classifier.

For the feature selection, we generated the word embeddings using Word2Vec, where each word is presented by a high dimension vector. Word2Vec is a pretrained statistical model for efficiently learning a standalone word embedding from a text corpus. It was developed by Google (Mikolov et al. 2013) and has become the de facto standard for developing pre-trained word embedding. The advantage of leveraging this model is that it was built using billions of words with a vast corpus of language that captures word meanings in a statistically robust manner. The dimension of the vector space is 300. For each Tweet, the aggregated vector is weighted by the value of the TF-IDF.

In addition, we also considered social media content-based features. The posted time is a critical feature because the valid "presence" posts must be made within the range of opening time. Many Tweets are synchronized from other applications such as Facebook, Swarmapp, Artic, Foursquare, etc. The domain name with the check-in app "Swarmapp" could have more probability for the people presence rather than art institutes application such as "artic." Therefore, whether the domain name is a check-in application name could be an important feature. The counts of favorites, retweets, hashtags, and mentions could also be essential features for identifying the features. When visiting and making a post in a museum, people may mention some official accounts and persons of significance to share the joy and findings. In addition to the aforementioned features, the username of the users could also be a critical feature. For example, some users are official accounts, and they would not normally make a check-in post. Therefore, we check if the usernames have strings such as "art," "archeo," "Chicago," "museum," etc. Finally, combining the word embeddings and the other selected Tweet-content-based features, we select 309-dimension vectors for each data point. Table 2 shows a summary of the selected features.

#### *2.2.3 Classifier formulation and performance evaluation*

In this step, we tested the performance of different categories of classifiers. We selected a traditional classifier, i.e., Support Vector Machine (SVM), an ensemble classifier of the Random Forest, and the shallow neural network (that contains three types of layers). The training/testing data ratio is 8:2.

The evaluation metrics of the text classifiers measure the performance of making the right classification decision from different methods. Generally, four metrics are widely used: accuracy, precision, recall, F1-score based on the number of true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN), as illustrated in Eqs. (1)–(4). The significance of these four elements may vary based on the classification application. It is noted that compared to the accuracy, the last three metrics are more

**Table 2** Summary of the selected features

| Categories | Selected features | Type | Dimension num. |
|---|---|---|---|
| Word embedding | 300-dimension word embeddings weighted by the TF-IDF | Float | 300 |
| Tweet-content | Whether the posted time is within opening hours | Binary | 1 |
| | Whether the username of users contain the keywords like "art" | Binary | 1 |
| | Whether the domain name in the URL contains check-in apps | Binary | 1 |
| | Hashtags in the Tweet count | Integer | 1 |
| | Mentions in the Tweet count | Integer | 1 |
| | Favorite/like count | Integer | 1 |
| | Retweet count | Integer | 1 |
| | Posted hour | Integer | 1 |
| | Posted day of the week | Integer | 1 |

meaningful in terms of the effectiveness of the text classifiers because the accuracy is insensitive to variations in the number of correct decisions due to the large value of the numerator (TP+TN) (Kowsari et al. 2019).

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F1-score} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{4}$$

The performance metrics of different classifiers from this case study are listed in Table 3 in terms of accuracy, precision, recall, and F1-score. It can be seen that the accuracy of the different classifiers is in a similar range, with the Random Forest and the neural network slightly being higher. This is as expected because we have a large number of true negatives when calculating the accuracy using Eq. (1).

As mentioned in the last section, the precision and recall are more meaningful in the evaluation of the effectiveness of the text classifiers. Although the Random Forest has a relatively high score of precision, it has a lower score of the recall score. This means the classification algorithm could not recognize the "presence" of the user and label it as the "not present." Since we need to know the number of the valid presence of the people in buildings, it is desirable to see a higher recall score. In terms of the F1-score, the neural network performs the best with a score of 0.8. F1-score is an overall metric combining the precision and the recall. It can be seen that the neural network performs slightly better than the other two classifiers.

**Table 3** Summary of the performance metrics of different classifiers

| Performance metric | SVM | Random Forest | Shallow neural network |
|---|---|---|---|
| Accuracy | 0.8485 | 0.9091 | 0.9091 |
| Precision | 0.6000 | 0.8333 | 0.7500 |
| Recall | 0.8571 | 0.7143 | 0.8571 |
| F1-score | 0.7059 | 0.7692 | 0.8000 |

In Table 4, the detailed classification results of the testing sets are presented using the shallow neural network. The labels "1" and "0" represent the status of the people-present (positive) and people-not-present (negative). Majorities of the labels belong to be "0" (i.e., people not present). For the labels "1", the results show that the method could basically distinguish them from most of the "0" labels (not-present labels). Indexes 59 and 168 were mislabeled, but their prediction scores are above 0.1. In addition, Index 96 was mislabeled to be "1" although they should be "0". The Tweets that are easy to be semantically differentiated, such as Indexes 198 and 30, have a high prediction score.

#### *2.2.4 Typical occupancy schedule extraction results from Tweeter*

We need to translate the count of classified "presence" Tweet to the building occupancy information. There exist sources of uncertainties in this translation. For example, we might not know how many hours people will stay there if they only have one valid "presence" post. Even if they have several posts, we are still not confident about how long he/she will stay. Therefore, we propose the following rules to extract the building occupancy pattern:

- Use one hour as a time slot.
- Count one person if the posts indicate the presence from the classifier.
- Assume probability > 0.5 as presence.
- Assume every person's average duration in the place from Google Maps statistical information. For example, we get the information from Google Maps that normally people will stay in the Art Institute of Chicago for up to three hours. A Monte Carlo simulation could be conducted to allow for the uncertainty of the people stay time in the place. However, for this feasibility study, we use the fixed average stay duration, as suggested by the Google Maps.
- If a person has two valid posts within several hours, we assume his/her presence in these several hours.

On top of that, we add up the count of the classified presence tweet in the same time slot in each weekday for all the historical data. The extracted occupancy pattern is aggregated time series curves for different weekdays. In this

**Table 4** Demonstration of the classification results using the shallow neural network

| Index | Posted Time | Tweets | Label | Prediction | Score |
|---|---|---|---|---|---|
| 136 | 4/1/2019 3:20 | Art Institute of Chicago will be hosting Gregg... | 0 | 0 | 0.00257 |
| 139 | 3/19/2019 4:03 | The Art Institute of Chicago is hosting Every... | 0 | 0 | 0.02752 |
| 198 | 2/8/2019 11:27 | I'm at The Art Institute of Chicago - @ artins... | 1 | 1 | 0.60308 |
| 59 | 3/12/2019 15:38 | Art Institute was amazing! # rembrandt # beaut... | 1 | 0 | 0.16830 |
| 96 | 3/30/2019 16:43 | Cut Piece, de Djanira. Performance, Art Instit... | 0 | 1 | 0.63309 |
| 23 | 3/27/2019 10:48 | Hopper @The Art Institute of Chicago https://w... | 1 | 1 | 0.50097 |
| 30 | 3/28/2019 13:38 | I'm at The Art Institute of Chicago - @ artins... | 1 | 1 | 0.73036 |
| 54 | 3/20/2019 5:10 | Can't wait to see this babe in May. # wcw # tr... | 0 | 0 | 0.00174 |
| 39 | 4/7/2019 16:01 | Got to spend an afternoon this weekend with Va... | 1 | 1 | 0.70794 |
| 66 | 4/4/2019 21:00 | Criticized for Failing to Consult Indigenous G... | 0 | 0 | 0.02370 |
| 67 | 3/28/2019 8:21 | Thanks so much for this Art Institute of Chica... | 0 | 0 | 0.05156 |
| 88 | 3/25/2019 14:43 | Wall-Floor Positions, de Gustave Klimt. Video... | 0 | 0 | 0.04427 |
| 63 | 4/12/2019 8:10 | School of the Art Institute of Chicago has nam... | 0 | 0 | 0.27143 |
| 168 | 3/14/2019 15:15 | A. Lincoln # artinstituteofchicago # chicago #... | 1 | 0 | 0.48824 |
| 86 | 3/24/2019 9:43 | Autorretrato aos 13, de Giotto. Desenho, Art I... | 0 | 0 | 0.04016 |
| 184 | 3/23/2019 1:00 | The Art of Reading at the Art Institute of Chi... | 0 | 0 | 0.01482 |
| 55 | 4/4/2019 10:16 | Art Institute of Chicago delayed exhibition of... | 0 | 0 | 0.00174 |
| 25 | 3/27/2019 11:33 | Art museum I'm ready to come home tbh. Work to... | 1 | 1 | 0.65585 |
| 72 | 3/13/2019 0:13 | I Like America and America Likes Me, de Alexan... | 0 | 0 | 0.00103 |
| 158 | 4/9/2019 21:12 | Art Institute of Chicago where Swami Ji delive... | 0 | 0 | 0.01881 |
| 60 | 4/3/2019 14:40 | In a move museum leadership is calling unprec... | 0 | 0 | 0.07430 |
| 110 | 4/9/2019 20:13 | I'm too sad to tell you, de Joseph Beuys. Vide... | 0 | 0 | 0.03527 |
| 199 | 3/18/2019 11:22 | @ JohnMu Just about every result page for the ... | 0 | 0 | 0.03667 |

way, we obtained the weekly typical occupancy schedules at the building level to be used as inputs for the building energy models, as shown in Figure 4, which shows two different types of typical occupancy daily schedules. It is noted that the opening hour is 10:30 to 20:00 on Thursday while 10:30 to 17:00 on Friday.

## 2.3 Web scraping from Facebook and Google Maps

The second approach is to utilize the processed GPS location tracking data provided by social network makers such as Facebook and Google Maps. Web scraping techniques are used to obtain the data and extract the typical occupancy schedules at the building level.

Figure 5 shows the sample of "Popular Times" by Google Maps and "Popular Hours" by Facebook. The principle behind these types of data lies in that these social network giants use aggregated and anonymized data from users who have opted in to share their real-time location. These companies also have Points-of-Interest (POI) building footprints (polygons), which determine the location, shape, and size of a place. Based on these data, machine-learning algorithms are used to join the GPS data against the building footprints to derive the occupancy information.

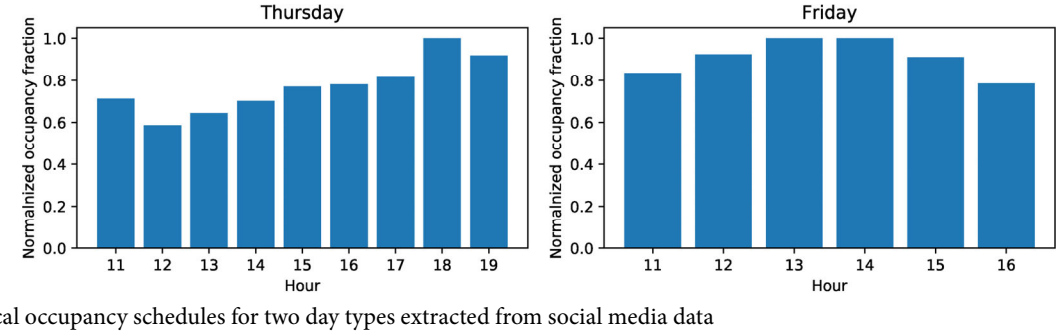
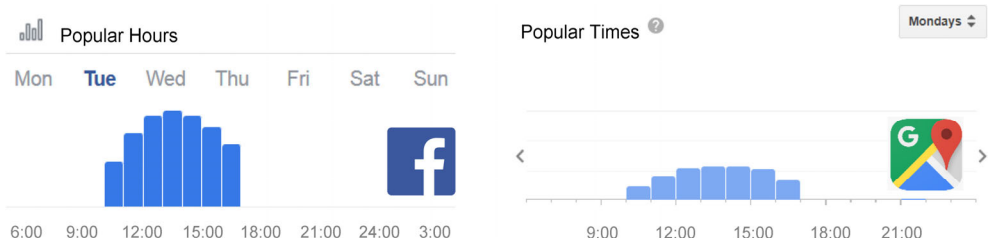



**Fig. 4** Typical occupancy schedules for two day types extracted from social media data




(a) Facebook Popular Hours (b) Google Maps Popular Times

**Fig. 5** The sample of "Popular Times" by Google Maps and "Popular Hour" by Facebook

Please note that the data from Facebook and Google Maps are relative occupancy information (i.e. normalized occupancy fraction for a given day) that is depicted as the height of the bar as shown in Figure 5. Therefore, we directly scraped the data from their websites. Figure 6 depicts the bar chart of the extracted typical building occupancy schedules for the Art Institute of Chicago from Facebook and Google Maps. The extracted occupancy schedules from Facebook and Twitter have a similar trend, but there still exist deviations. The deviations lie in that there might be several users who would be visiting who do not have Google Maps or location history enabled.

## 2.4 Results and discussion

It can be seen from Figure 7 that the extracted building occupancy schedules from different data sources (Twitter, Facebook, and Google Maps) share a similar trend but slightly distinct from each other. Figure 8 further calculates the Pearson correlation coefficient (PCC), which measures the strength and direction of the relationship between two variables, for these occupancy schedules extracted from two given approaches. The PCC has a value between +1 and −1, where 1 represents a total positive linear correlation, 0 indicates that there is no linear correlation, and −1 gives a total negative linear correlation. All PCCs between −0.8 and +0.8 are considered not significant. The correlation coefficients between Facebook and Google Maps achieve a high score (~0.95), while the value between Twitter and Facebook/Google Maps is slightly lower. This observation requires further validation and corrections to consider the underlying uncertainties. For the approach 1 (i.e., text classification from Tweets) in Section 2.2, it is believed to have more uncertainties associated with algorithms used

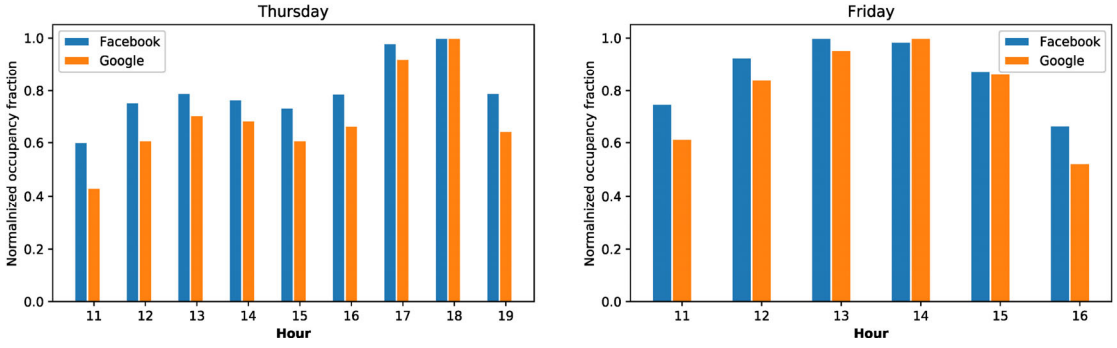



**Fig. 6** Comparisons of typical occupancy schedules for two days extracted from Facebook and Google Maps

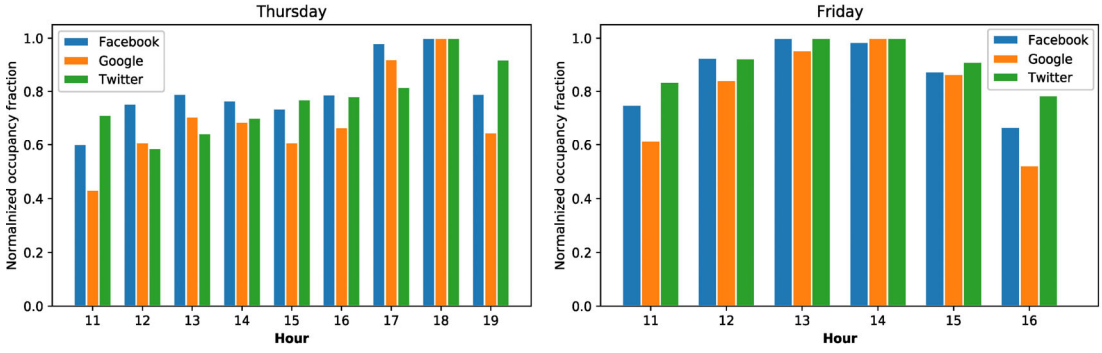



**Fig. 7** Comparisons of typical occupancy schedules for two days extracted from two approaches

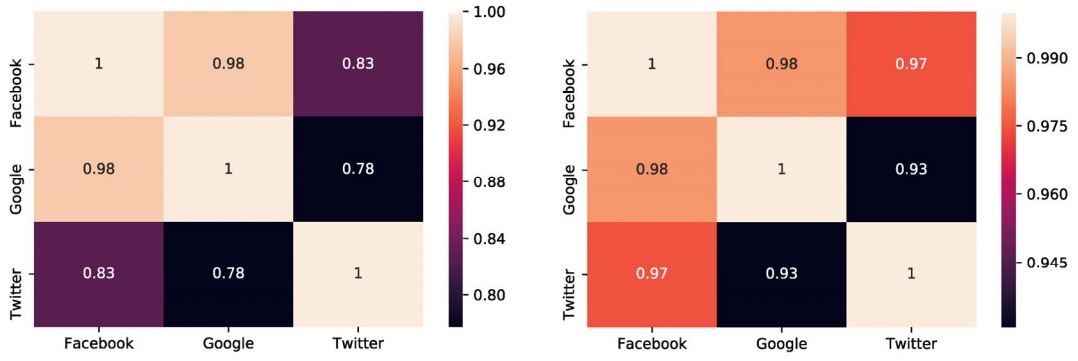



**Fig. 8** Heat map of Pearson correlation coefficient for schedules extracted from two approaches (left map: Thursday; right map: Friday)

during the entire procedure. For example, the Twitter posts may occur within the normal hours of operation but not occur at the time of occupancy, which results in inaccurate timestamp data. For the approach 2 (i.e., web-scraping from Facebook/Google Maps) in Section 2.3, the uncertainty arises from the fact that users who would be visiting might not have Google Maps or location history enabled. Furthermore, the ratio of users to non-user of social media should be acknowledged due to the age bracket of the social media users. Such uncertainty will lead to inaccurate extracted occupancy schedules for both approaches.

## 3 Integration TOSSM with building energy modeling: case study 2 for a university museum

In this section, a case study for a university museum is presented to demonstrate the application of the extracted TOSSM into building energy models. Another objective of this case study is to verify whether the additional occupancy features from social media could improve the prediction performance of the building energy model. Therefore, data-driven building energy models are established for the hourly cooling and heating energy prediction with or without the social media extracted occupancy features. Section 3.1 discusses the data preprocessing and Section 3.2 describes the feature selection process. Section 3.3 details the construction of the data-driven models. In Section 3.4, the results and discussion are presented.

### 3.1 Data preprocessing

Alabama Museum of Natural History, the case study building in this section, is located in Smith Hall at the University of Alabama campus in Tuscaloosa, AL. This building is selected as the case study building because we have detailed and sufficient data for the model development and validation (e.g., the building floor plan, building system configuration, energy usage data, etc.), as well as the actual meteorological data from an onsite weather station. The floor plans of the museum and the location of the air handling unit (AHU) are depicted in Figure 9. The chilled water and hot water are from a campus energy plant through a district network. The weather data for this case study is collected from an onsite weather station on the campus, which is about 120 meters southeast of the Smith Hall, to accurately capture relevant microclimate variation. The weather data is logged in a two-minute time step and it is further resampled to an hourly mean time series. As Table 5 exemplifies, the weather data has the attributes of dry bulb temperature, relative humidity, dew point temperature, wind speed, gust speed, wind direction, and global solar radiation. The typical occupancy schedules at the building level are extracted using approach 2 (i.e., web-scraping from Google Maps) described in Section 2.3, as depicted in Figure 10. The chilled water (CHW) usage and hot water (HW) usage are metered in a 15-minute interval. We select the data at a time frame from March 28$^{th}$, 2018 to May 23$^{rd}$, 2019, and further process these data into an hourly time series. Figure 11 depicts a weekly example of the energy usage data for both chilled water and hot water consumption. Both temperatures and humidity need to be controlled in this building, and a traditional cool-reheat approach was used for dehumidification. This explains the relatively high hot water consumption in August in a humid climate zone.

### 3.2 Filter-method-based feature selection

Selecting a set of correlated input features is critical for building a data-driven building energy prediction model. The input features can be categorized into exterior factors such as meteorological data, internal factors such as occupancy, HVAC operation data from building automation systems, and time-lag history data, etc. (Zhang and Wen 2019). Based on the aforementioned factors and the input data availability for this case study, the raw input features we consider include the meteorological data such as dry bulb temperature, relative humidity, dew point temperature, solar radiation, wind speed, gust speed, wind direction; the

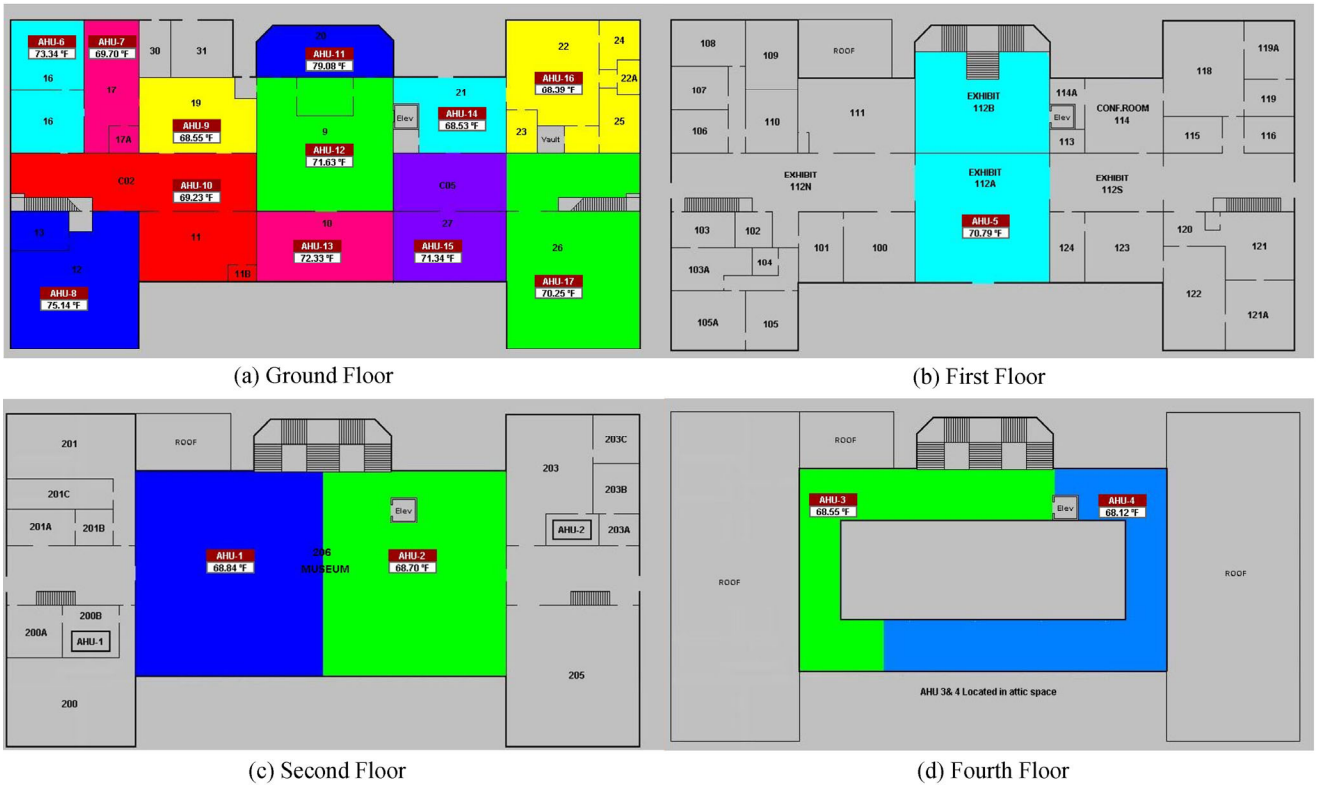



**Fig. 9** Floor plan and AHU location of Alabama Museum of Natural History

**Table 5** Example of weather station data set used in an hourly basis

| Time (CDT) | Dry bulb temperature (°F) | Relative humidity (%) | Dew temperature (°F) | Wind speed (mph) | Gust speed (mph) | Wind direction (°) | Solar radiation (W/m$^2$) |
|---|---|---|---|---|---|---|---|
| 3/29/2018 11:00 | 67.64 | 80.69 | 61.34 | 3.55 | 9.37 | 143.77 | 81.63 |
| 3/29/2018 12:00 | 59.92 | 94.34 | 58.32 | 1.74 | 5.67 | 214.93 | 50.17 |
| 3/29/2018 13:00 | 60.09 | 95.32 | 58.78 | 0.38 | 2.18 | 220.17 | 56.97 |
| 3/29/2018 14:00 | 60.20 | 95.06 | 58.82 | 0.23 | 2.09 | 261.27 | 48.23 |
| 3/29/2018 15:00 | 59.49 | 95.38 | 58.21 | 1.53 | 4.02 | 264.57 | 35.27 |
| 3/29/2018 16:00 | 59.57 | 96.30 | 58.54 | 4.22 | 7.67 | 210.27 | 29.57 |
| 3/29/2018 17:00 | 60.05 | 96.26 | 59.02 | 0.63 | 2.00 | 160.30 | 34.73 |

calendar features such as hour of day, day of week, day type, and month of year; the occupancy information such as the extracted building occupancy schedules from social media. It is noted that calendar features such as the hour of the day and day type could also indicate the occupancy condition and pattern (Wang and Srinivasan 2017; Wang et al. 2019) and may have a correlation with the occupancy features we extracted.

To determine the prominent features and improve the performance of the data-driven models, a filter-method-based feature selection approach reported in Ref. (Zhang and Wen 2019) is adopted. In this approach, the Pearson correlation coefficients (PCC) are calculated between each

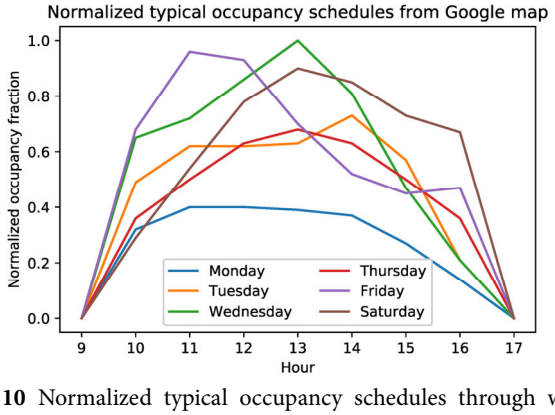



**Fig. 10** Normalized typical occupancy schedules through web-scrapping from Google Maps for Smith Hall

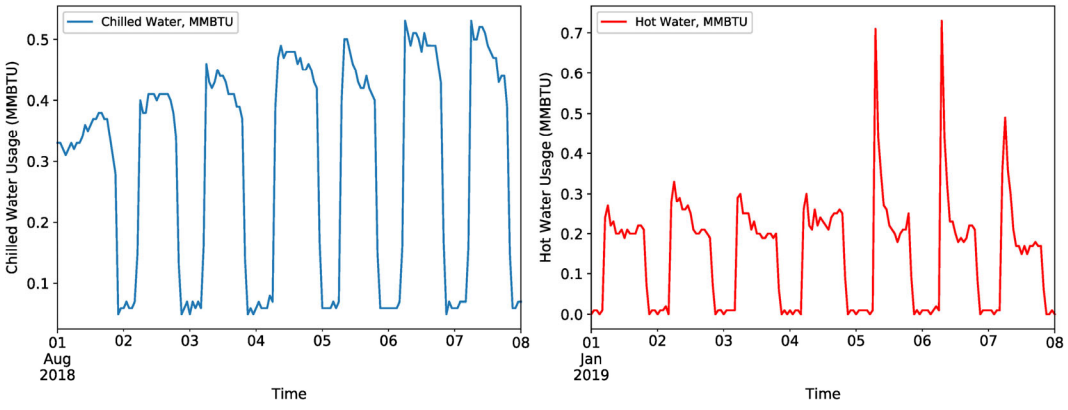



**Fig. 11** Weekly time series for chilled water usage and hot water usage in Smith Hall

input feature and the output of interest to filter out the weakly correlated features. Table 6 lists the results from this feature selection method. It can be seen that dry bulb temperature, dew point temperature, solar radiation, occupancy, and relative humidity are the top five correlated features with the outputs. In comparison, the day of the week, gust speed, wind speed, and wind direction are the least four uncorrelated features with the output. The cut-off thresholds need to be carefully determined since the feature that has a low correlation with the output by itself can still provide a significant performance improvement when being combined with other features. From our domain knowledge, we keep the feature Day of Week since it may improve the prediction performance when being combined with other calendar features. However, the feature Wind Direction is eliminated in this step based on the feature selection results and our domain knowledge. On top of that, the Pearson correlation coefficients between the input features are also calculated to eliminate the features that are closely co-related. The results show that the features Wind Speed and Gust Speed are closely correlated (PCC>0.95). Since the feature Gust Speed shows a closer correlation with the output, this feature is kept, and the feature Wind Speed is eliminated.

Therefore, the input feature sets after the feature selection are composed of dry bulb temperature, dew point temperature, solar radiation, occupancy, relative humidity, hour of day, month of year, day of week, gust speed. It is noted that the occupancy feature might serve as a critical factor that contributes to the prediction improvement from the result in this section. To further investigate the efficacy of the occupancy feature extracted from social media, feature assessment of feature "Occupancy" is conducted in the next section.

**Table 6** Pearson correlation coefficients between each input feature and the output of interest

| Input features | Pearson correlation coefficient | |
|---|---|---|
| | CHW usage | HW usage |
| Dry bulb temperature | 0.7658 | −0.2507 |
| Dew point temperature | 0.6644 | −0.332 |
| Solar radiation | 0.4579 | 0.1913 |
| Occupancy | 0.2576 | 0.1990 |
| Relative humidity | −0.2069 | −0.1437 |
| Hour of day | 0.13 | 0.0376 |
| Month of year | 0.142 | −0.0207 |
| Day of week | −0.0268 | −0.0659 |
| Gust speed | 0.0886 | 0.157 |
| Wind speed | 0.0497 | 0.1237 |
| Wind direction | −0.0469 | 0.0265 |

## 3.3 Feature importance assessment for feature "Occupancy"

In this section, two feature input sets are compared by two different machine-learning algorithms to evaluate the feature importance of feature "Occupancy". As Table 7 shows, the features in feature input set 1 are derived from the result in Section 3.3, while the feature input set 2 has the same features with the feature "Occupancy" removed. Based on that, eight data-driven models are constructed with different feature input sets, different machine-learning algorithms, and different outputs of interest, as shown in Table 8.

Two well-used machine-learning algorithms, Random Forest (RF) (Liaw and Wiener 2002) and XGBoost (XGB)

**Table 7** Two feature sets for evaluating feature "Occupancy"

| Set | Features |
|---|---|
| Feature input set 1 | Dry bulb temperature, dew point temperature, solar radiation, relative humidity, hour of day, month of year, day of week, gust speed |
| Feature input set 2 | Dry bulb temperature, dew point temperature, solar radiation, **occupancy**, relative humidity, hour of day, month of year, day of week, gust speed |

**Table 8** Description of eight constructed models

| Case number & name | Features | Machine learning algorithms | Output |
|---|---|---|---|
| Case 1: CHW-RF-w/Occ | Feature input set 1 | Random Forest (RF) | Hourly Chilled Water Usage |
| Case 2: CHW-RF-w/oOcc | Feature input set 2 | Random Forest | Hourly Chilled Water Usage |
| Case 3: CHW-XGB-w/Occ | Feature input set 1 | XGBoost (XGB) | Hourly Chilled Water Usage |
| Case 4: CHW- XGB -w/oOcc | Feature input set 2 | XGBoost | Hourly Chilled Water Usage |
| Case 5: HW-RF-w/Occ | Feature input set 1 | Random Forest | Hourly Hot Water Usage |
| Case 6: HW-RF-w/oOcc | Feature input set 2 | Random Forest | Hourly Hot Water Usage |
| Case 7: HW- XGB-w/Occ | Feature input set 1 | XGBoost | Hourly Hot Water Usage |
| Case 8: HW- XGB -w/oOcc | Feature input set 2 | XGBoost | Hourly Hot Water Usage |

(Chen et al. 2015), are adopted as the prediction models. Random forest regression model utilizes an ensemble learning method that operates by constructing a multitude of decision trees at training time and outputting the mean prediction of the individual trees (Wang et al. 2018), which achieves a significant improvement in terms of accuracy and stability compared to the basic decision trees. XGBoost, proposed in 2014, is an implementation of gradient boosted decision trees designed for speed and performance (Chen and Guestrin 2016). This algorithm has recently been dominating applied machine learning and Kaggle competitions for structured or tabular data. To form an optimal model architecture, a set of hyperparameters needs to be learned and tuned (Duan et al. 2003). The main parameters affecting the RF performance include number of trees (NT), the maximum depth of the tree (MDT), the maximum number of features (MNF), the minimum number of samples required to split a node (MSS), and the minimum number of samples required at each leaf node (MSL). Likewise, XGBoost has critical parameters such as number of trees (NT), the maximum depth of decision trees (MDT), learning rate, subsample number, etc.

Regarding the evaluation of the constructed prediction model, various performance indicators are used, as shown in Eq. (5)–Eq. (8). They are mean absolute error (MAE), *R*-squared ($R^2$), the coefficient of variation of root-mean-squared error (CV (RMSE)), and normalized mean bias error (NMBE).

MAE reflects the average over the test sample of the absolute differences between prediction and actual observations where all individual differences have equal weight. $R^2$ and CV(RMSE) both indicate the goodness of fit for the prediction results with respect to the real data. It is noted that $R^2$ focuses more on the error observed over individual data points while CV(RMSE) quantifies the average error. The metric of NMBE indicates the error bias (positive or negative). Though NMBE could be a misleading metric for the prediction alone since the positive bias and negative bias may cancel out, it helps to present the relative position of the simulated data with respect to the measured data. In ASHRAE Guideline 14 (ASHRAE 2018), it suggests the error tolerance limits for building energy prediction, the CV(RMSE) and NMBE should be within 30% and ±10% for the hourly prediction data, respectively.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} \left| y_i - \hat{y}_i \right|}{n} \tag{5}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left( y_i - \hat{y}_i \right)^2}{\sum_{i=1}^{n} \left( y_i - \overline{y} \right)^2} \tag{6}$$

$$\mathrm{CV(RMSE)} = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n} \left( y_i - \hat{y}_i \right)^2}}{\overline{y}} \times 100\% \tag{7}$$

$$\mathrm{NMBE} = \frac{y_i - \hat{y}_i}{(n-1)\overline{y}} \times 100\% \tag{8}$$

where $\hat{y}_i$, $y_i$, and $\overline{y}$ represent the measured data, predicted data, and mean of the measured data, respectively; $n$ is the total number of the data samples.

### 3.4 Results and discussion

In this case study, these two data-driven models are implemented using the sklearn module in Python (v3.7). The training and testing datasets are hourly data from Mar 28th, 2018 to May 23rd, 2019, as described in Section 3.1. The ratio between the training and testing datasets is 9:1 for both the chilled water and hot water usage prediction. For the hyperparameter tuning, the grid search, random search (Bergstra and Bengio 2012), and Bayesian optimization algorithms (Klein et al. 2016) are commonly used methods. In this study, the random search approach is used with 3-fold cross-validation. Table 9 details the hyperparameter setting of the models to ensure them well-configured.

Table 10 compares the model performance for cases 1–4 with chilled water usage predictions. It can be seen that both CV (RMSE) and NMBE are bounded within 30% and

**Table 9** Hyperparameter setting of the models

| Case number & name | Hyper-parameters setting |
|---|---|
| Case 1: CHW-RF-w/Occ | n_estimators=1400, max_depth=90, min_samples_split=2, min_samples_leaf=1, max_features=sqrt, bootstrap=False |
| Case 2: CHW-RF-w/oOcc | n_estimators=1400, max_depth=90, min_samples_split=2, min_samples_leaf=1, max_features=sqrt, bootstrap=False |
| Case 3: CHW-XGB-w/Occ | learning_rate=0.1, n_estimators=300, max_depth=7, min_child_weight=9, gamma=0, subsample=0.9, colsample_bytree=0.9 |
| Case 4: CHW- XGB -w/oOcc | learning_rate=0.1, n_estimators=300, max_depth=7, min_child_weight=1, gamma=0, subsample=0.8, colsample_bytree=0.8 |
| Case 5: HW-RF-w/Occ | n_estimators=1400, min_samples_split=2, min_samples_leaf=1, max_features=sqrt, max_depth=,60 bootstrap=False |
| Case 6: HW-RF-w/oOcc | n_estimators=1400, min_samples_split=2, min_samples_leaf=1, max_features=sqrt, max_depth=60, bootstrap=False |
| Case 7: HW- XGB-w/Occ | learning_rate=0.1, n_estimators=350, max_depth=13, min_child_weight=5, gamma=0, subsample=0.9, colsample_bytree=0.7 |
| Case 8: HW- XGB -w/oOcc | learning_rate=0.1, n_estimators=400, max_depth=7, min_child_weight=7, gamma=0, subsample=0.9, colsample_bytree=0.7 |

**Table 10** Comparison of cases 1–4 for chilled water usage prediction in Smith Hall

| | Case 1 CHW-RF-w/Occ | Case 2 CHW-RF-w/oOcc | Improvement ratio (%) | Case 3 CHW-XGB-w/Occ | Case 4 CHW-XGB-w/oOcc | Improvement ratio (%) |
|---|---|---|---|---|---|---|
| MAE (MMBTU) | 0.0263 | 0.0275 | 4.36 | 0.0256 | 0.0278 | 7.91 |
| $R^2$ | 0.9155 | 0.909 | 0.72 | 0.9192 | 0.9111 | 0.89 |
| CV(RMSE) (%) | 4.64 | 4.81 | 3.53 | 4.53 | 4.76 | 4.83 |
| NMBE (%) | −0.96 | −1.04 | 7.74 | 0.4267 | 1.0268 | 58.44 |

±10% for all the four cases, which indicates a good prediction performance. Comparing case 1 & case 2 and case 3 & case 4, it is evident that the model prediction performance is slightly enhanced for both machine-learning algorithms with occupancy information extracted from social media. MAE and CV (RMSE) increase by ~5%, which is modest. The improvement of $R^2$ is limited, probably due to the co-linearity between the feature "Occupancy" and the three calendar features (hour of day, day of week, and month of year). Table 11 shows the model performance comparison of cases 5–8 with hot water usage predictions. Similar results can be found. Both CV(RMSE) and NMBE are bounded within 30% and ±10% for all the four cases. Comparing case 5 & case 6 and case 7 & case 8, the model prediction performance is slightly enhanced for both machine-learning algorithms except that the NMBE decreases slightly using RF (the absolute values of RF for both bases are small although the relative improvement ratio is high). MAE increase by ~5% for both algorithm and CV (RMSE) increases by 1.87% and 4.12% for RF and XGB, respectively. The improvement of $R^2$ is also limited.

Figure 12 depicts the comparison of predicted data and measured data in the testing set (first 200 data points) for chilled water usage and hot water usage considering the occupancy feature. Overall, the incorporation of the Feature "Occupancy" could improve the hourly energy usage prediction to a small extent regarding the four evaluation metrics. In other words, from an engineering perspective, the data-driven models without the integration of the TOSSM but with calendar features have already achieved a high prediction performance in this building type. The model performance improvement is rather modest probably, due to the following justifications. First, the case study building is a school museum. The number of occupants is normally lesser than its design value and will have less impact on the building loads. Furthermore, the HVAC control for

**Table 11** Comparison of cases 5–8 for hot water usage prediction in Smith Hall

| | Case 5 HW-RF-w/Occ | Case 6 HW-RF-w/oOcc | Improvement ratio (%) | Case 7 HW-XGB-w/Occ | Case 8 HW-XGB-w/oOcc | Improvement ratio (%) |
|---|---|---|---|---|---|---|
| MAE (MMBTU) | 0.0236 | 0.0248 | 4.84 | 0.0244 | 0.0256 | 4.69 |
| $R^2$ | 0.8883 | 0.8838 | 0.51 | 0.9001 | 0.8912 | 1.00 |
| CV(RMSE) (%) | 4.19 | 4.27 | 1.87 | 3.96 | 4.13 | 4.12 |
| NMBE (%) | 0.29 | 0.23 | −26.09 | 0.3538 | 0.5632 | 37.18 |

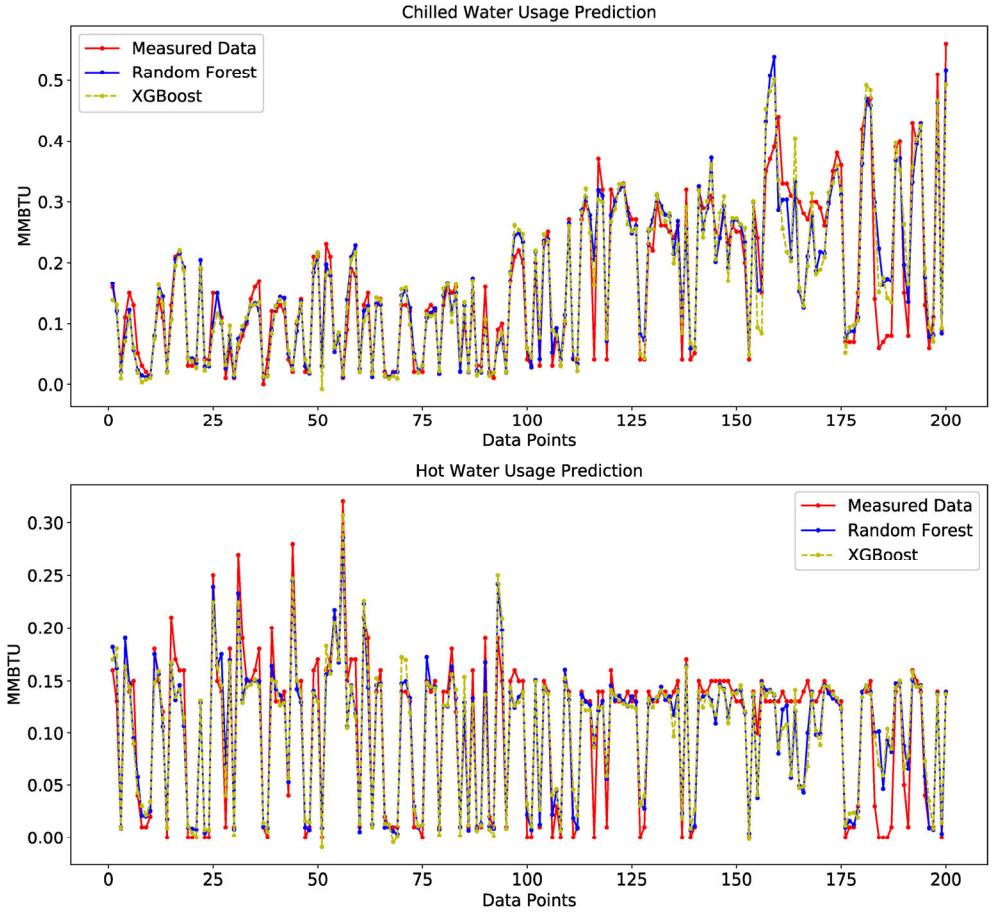



**Fig. 12** Prediction performance comparison for chilled water usage and hot water usage considering occupancy feature

this 110-year old building with a typical schedule-based operation is not heavily occupant-centric. Therefore, the influence of the occupancy might not be as considerable as that in a modern large office building with occupant-centric controls. Secondly, the Feature "Occupancy" has a strong correlation to the calendar features. It is evident from cases 2, 4, 6, and 8 that using calendar features as the indictor of occupancy has already achieved a sufficient accuracy for this type of building.

## 4 Conclusions, limitations, and future work

In this paper, we presented two approaches to extract the typical occupancy schedules for the input to the building energy simulation using social media data. The first approach formulates a semantic classifier to identify whether people are present in the space where they are posting from Twitter. With assumed people counting rules, the typical occupancy schedules are then extracted. In the second approach, web scraping techniques are used to extract the building typical occupancy schedules based on the processed GPS tracking data provided by social network makers such as Facebook and Google Maps. The quantitative results show that the extracted building occupancy schedules from three data sources (Twitter, Facebook, and Google Maps) share a similar trend but slightly distinct from each other, which requires further validation and corrections.

To further demonstrate the application of the extracted typical occupancy schedules from social media (TOSSM), data-driven models for hourly energy usage prediction of a university museum are developed using Random Forest and XGBoost, with the integration of the TOSSM. For the chilled water usage prediction, MAE and CV(RMSE) increase by ~5% while the improvement of $R^2$ is limited. NMBE increases by 58.44% for XGBoost, but the absolute increase value is small. Similar results can be observed for the hot water usage prediction. By comparing the models with and without the occupancy schedule features, the incorporation of the TOSSM could improve the hourly energy usage prediction to a certain extent regarding the four adopted evaluation metrics.

The future work includes improving, validating, and correcting the occupancy schedule estimation from two proposed approaches using the visitor counting (e.g., ticket information, people counting data from occupancy sensors, etc.) from the museum. We will investigate some uncertainties

that mentioned in Section 2.4:

- The inaccurate timestamp issue, the fake account issue, etc., for approach 1 (text classification through Tweets) in Section 2.2.
- The fact that users who would be visiting might not have Google Maps or location history enabled for approach 2 (web-scraping from Facebook/Google Maps) in Section 2.3.
- Considering the non-user of social media for both approaches.

We would also like to implement the evaluation of the value proposition of using the TOSSM for building energy modeling, as described in Section 3, for different types of buildings, such as office buildings, school buildings, hotel buildings, etc. It is anticipated that building energy consumption in some of these building types will be more correlated with occupancy schedules. Furthermore, such occupancy information at the building level will be incorporated with the urban-scale community and city energy modeling.

## Acknowledgements

The authors would like to thank the personnel at the facility energy management team of the University of Alabama, especially Mr. Greg McKelvey and Mr. Donnie Grill, for their assistance in providing the data. This study is supported by NSF project #1827757 "PFI-RP: Data-Driven Services for High Performance and Sustainable Buildings.

## References

Abergel T, Dean B, Dulac J (2017). Towards a zero-emission, efficient, and resilient buildings and construction sector: Global Status Report 2017. UN Environment and International Energy Agency, Paris, France.

Agarwal Y, Balaji B, Gupta R, Lyles J, Wei M, Weng T (2010). Occupancy-driven energy management for smart building automation. In: Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building—BuildSys'10, Zurich, Switzerland.

ASHRAE (2018). ASHRAE Guideline. Guideline 16-2010. Selecting Outdoor, Return, and Relief Dampers for Air-Side Economizer Systems.

Bentz JO, Rigg BD, Gillette TN, Tasker TR, McCune T, Uerkvitz JW, Atchison SB, Middleton DS, Noor AM (2019). Thermostat with occupancy detection based on social media event data, Google Patents.

Bergstra J, Bengio Y (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(1): 281–305.

Bojanowski P, Grave E, Joulin A, Mikolov T (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5: 135–146.

Chen T, He T, Benesty M, Khotilovich V, Tang Y (2015). Xgboost: Extreme gradient boosting. R package version 0.4-2: 1–4.

Chen T, Guestrin C (2016). Xgboost: A scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference On Knowledge Discovery and Data Mining, San Diego, CA , USA.

Christensen K, Melfi R, Nordman B, Rosenblum B, Viera R (2014). Using existing network infrastructure to estimate building occupancy and control plugged-in devices in user workspaces. *International Journal of Communication Networks and Distributed Systems*, 12: 4.

Dong B, Prakash V, Feng F, O'Neill Z (2019). A review of smart building sensing system for better indoor environment control. *Energy and Buildings*, 199: 29–46.

Duan K, Keerthi SS, Poo AN (2003). Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing*, 51: 41–59.

Gu J, Xu P, Pang Z, Chen Y, Ji Y, Chen Z (2018). Extracting typical occupancy data of different buildings from mobile positioning data. *Energy and Buildings*, 180: 135–145.

Henrique J (2019). GetOldTweets3: A Python 3 library and a corresponding command line utility for accessing old tweets.

Jazizadeh F, Jung W (2018). Personalized thermal comfort inference using RGB video images for distributed HVAC control. *Applied Energy*, 220: 829–841.

Jin M, Bekiaris-Liberis N, Weekly K, Spanos CJ, Bayen AM (2018). Occupancy detection via environmental sensing. *IEEE Transactions on Automation Science and Engineering*, 15: 443–455.

Jung W, Jazizadeh F (2019). Human-in-the-loop HVAC operations: A quantitative review on occupancy, comfort, and energy-efficiency dimensions. *Applied Energy*, 239: 1471–1508.

Klein A, Falkner S, Bartels S, Hennig P, Hutter F (2016). Fast Bayesian optimization of machine learning hyperparameters on large datasets. arXiv preprint arXiv: 1605.07079.

Kowsari K, Jafari Meimandi K, Heidarysafa M, Mendu S, Barnes L, Brown D (2019). Text classification algorithms: A survey. *Information*, 10(4): 150.

Liaw A, Wiener M (2002). Classification and regression by randomForest. *R News*, 2(3): 18–22.

Lu X, Feng F, O'Neill Z (2019). Acquisition of Typical Occupancy Schedules for Commercial Buildings from Social Networks. New York: Association for Computing Machinery.

Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013). Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems.

Muroni A, Gaetani I, Hoes P-J, Hensen JLM (2019). Occupant behavior in identical residential buildings: A case study for occupancy profiles extraction and application to building performance simulation. *Building Simulation*, 12: 1047–1061.

Naylor S, Gillott M, Lau T (2018). A review of occupant-centric building control strategies to reduce building energy use. *Renewable and Sustainable Energy Reviews*, 96: 1–10.

Newsham GR, Xue H, Arsenault C, Valdes JJ, Burns GJ, Scarlett E, Kruithof SG, Shen W (2017). Testing the accuracy of low-cost data streams for determining single-person office occupancy and their use for energy reduction of building services. *Energy and Buildings*, 135: 137–147.

Pang Z, Xu P, O'Neill Z, Gu J, Qiu S, Lu X, Li X (2018). Application of mobile positioning occupancy data for building energy simulation: An engineering case study. *Building and Environment*, 141: 1–15.

Pang Z, Chen Y, Zhang J, O'Neill Z, Cheng H, Dong B (2020). Nationwide energy saving potential evaluation for office buildings with occupant-based building controls. Paper presented in 2020 ASHRAE Winter Meeting, Orlando, FL, USA.

Park JY, Ouf MM, Gunay B, Peng Y, O'Brien W, Kjærgaard MB, Nagy Z (2019). A critical review of field implementations of occupant-centric building controls. *Building and Environment*, 165: 106351.

Pennington J, Socher R, Manning C (2014). Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).

Samuelson HW, Ghorayshi A, Reinhart CF (2016). Analysis of a simplified calibration procedure for 18 design-phase building energy models. *Journal of Building Performance Simulation*, 9: 17–29.

Sims KM, Weber EM, Bhaduri BL, Thakur GS, Resseguie DR (2017). Application of social media data to high-resolution mapping of a special event population. In: Griffith D, Chun Y, Dean D (eds), Advances in Geocomputation. Cham, Switzerland: Springer. pp. 67–74.

Stewart R, Urban M, S. Duchscherer, Kaufman J, Morton A, Thakur G, Piburn J, Moehl J (2016). A Bayesian machine learning model for estimating building occupancy from open source data. *Natural Hazards*, 81: 1929–1956.

Stewart R, Piburn J, Weber E, Urban M, Morton A, Thakur G, Bhaduri B (2017). Can social media play a role in the development of building occupancy curves? In: Griffith D, Chun Y, Dean D (eds), Advances in Geocomputation. Cham, Switzerland: Springer. pp. 59–66.

Wallach HM (2006). Topic modeling: beyond bag-of-words. In: Proceedings of the 23rd International Conference on Machine Learning.

Wang Z, Srinivasan RS (2017). A review of artificial intelligence based building energy use prediction: Contrasting the capabilities of single and ensemble prediction models. *Renewable and Sustainable Energy Reviews*, 75: 796–808.

Wang Z, Wang Y, Zeng R, Srinivasan RS, Ahrentzen S (2018). Random Forest based hourly building energy prediction. *Energy and Buildings*, 171: 11–25.

Wang R, Lu S, Li Q (2019). Multi-criteria comprehensive study on predictive algorithm of hourly heating energy consumption for residential buildings. *Sustainable Cities and Society*, 49: 101623.

Wikipedia Contributors (2020). Tf–idf. Retrieved 03:44, February 16, 2020. Available at https://en.wikipedia.org/w/index.php?title=Tf%E2%80%93idf&oldid=938382443.

Wu HC, Luk RWP, Wong KF, Kwok KL (2008). Interpreting TF-IDF term weights as making relevance decisions. *ACM Transactions on Information Systems*, 26: 1–37.

Yu Z, Fung BC, Haghighat F, Yoshino H, Morofsky E (2011). A systematic procedure to study the influence of occupant behavior on building energy consumption. *Energy and Buildings*, 43: 1409–1417.

Zhang L, Wen J (2019). A systematic feature selection procedure for short-term data-driven building energy forecasting model development. *Energy and Buildings*, 183: 428–442.