UMM: Power Monitoring and Prediction Software

Yujing Song, Abenezer Monjor, Khondoker Prio (Ahnaf)
UMM CSci Department
Division of Science and Mathematics
University of Minnesota Morris
Morris, Minnesota 56267
songx827@morris.umn.edu,monjo003@morris.umn.edu,
yasin011@morris.umn.edu.

Abstract

UMM has long valued renewable energy and sustainability and has invested in solar cells and wind turbines. Since they both however heavily depend on time and weather conditions, the power harvested fluctuates. They produce, on average, about 60% of the electricity used on campus,but there are times when we produce more than the campus uses. When these specific occasions arise,the power harvested cannot fully be utilized and is sent back to the grid due to the lack of local storage. The rate we receive for contributing this excess electricity is typically substantially less than the rate at which the university buys power from the grid. We are working on a real time power monitoring system that compares the power consumption of the entire campus with the power from renewables. If, for instance, students had our access to our forecasting system on their phones, they could see when the campus is likely to be producing excess electricity and plan energy intensive activities like laundry etc.

1. UMM’s efforts for a better renewable and sustainable future

The University of Minnesota Morris (UMM) values renewable energy and sustainability. Within a mile or so of the campus there is biomass gasification, solar thermal and PV installations, geothermal, methane digestion, green buildings,, and more, all part of the campus’ goal to reduce UMM’s carbon footprint. A diverse and distributed renewable energy platform is a key part of morris’s plan to achieve carbon neutrality.

1.1 The Solar panels and Wind Turbines.

Two 1.65 megawatt wind turbines are situated just off campus at the University of Minnesota West Central Research and Outreach Center (WCROC).The wind turbines generate on average 60 percent of the campus’s electricity usage annually and can generate up to 100 percent on windy days.The turbines generate more than 10 million kWh of electricity per year.The turbine towers are about 230 feet high; each has three 135-foot blades.The turbines can produce electricity at wind speeds as low as 7.8 mph. Maximum electricity output occurs at 29 mph wind speeds.
A solar thermal array of locally produced panels heats the recreational pool at the Regional Fitness Center on campus. The 32 flat panels keep about 15 tons of CO_2 out of the atmosphere annually. There is also a 3kW photovoltaic demonstration solar array on campus outside of Science. One of the arrays tracks the sun, the other is fixed.

1.2 The power prediction system’s aim to aid existing efforts.

Since solar cells and wind turbines both heavily depend on time and weather conditions, the power harvested fluctuates. While we on average produce about 60% of the electricity used on campus, there are times when we produce more than the campus uses. When these specific occasions arise the power harvested cannot fully be utilized and is sent back to the grid. The cost for returning excess electricity to the grid is substantially less than the rate to purchase The University would prefer to adjust usage so that all produced electricity is used on campus.

A real time power monitoring system that compares the power consumption of the entire campus with the power produced through renewable sources will help determine when the power harvested is greater than the power consumed for the campus. This has several benefits. Students with an app on their phones could see that renewable energy can actually meet their entire power consumption needs and at times exceeds it (like during the peak hours). Real time graphs and visualizations could help increase demand for for solar cells and wind turbines so that renewable sources would suffice at all times of the day. It could also provide encouragement for behavior changing initiatives such as doing laundry or other power intensive activities during peak power production hours which would maximize the use of renewable power and minimize the need to send power back to the grid.

2. The Prediction Model

Our goal is to analyze the solar panels' energy data and create a model for predicting the production of solar panels. We used Rstudio to do all the model building steps

2.1 Using eGauge and UMM Weather Station Data.

There are two important data sources providing historical data. First is monthly UMM weather data summaries from the UMM campus website[1] Second is the solar panel production data for UMM Green Prairie Community from the Company eGauge for University of Minnesota Morris KW Solar[2].

Currently, we used multiple linear regression techniques to create model for predicting the production of solar panel energy in UMM Green Prairie (GP) in 2015. We assume that the conditions at the solar panels perfectly match the collected weather data despite issues with panel positions and time delays. We think this approximation is reasonable because the distance between solar panels and weather collectors is less than 0.3 miles.

Production data is measured in unit kilowatt-hours (KWH) and sampled every 15 minutes starting at 0:15 of January 1st in 2015, and ending at 0:00 of January 1st in 2016. We summed the production from all solar panel (3 solar panels in total) and called it *general production*. Production data shows negative number at night, because the solar panels were still working at night. We set those values to 0.

Our final linear model is as follows:

GenSP = 1.3038386 + 0.5613333*SolarEnergy + 0.0035486*TempOut - 0.0183855*HumOut

Where GenSP(KWH) is the expected value of general solar panels' energy, SolarEnergy(W/m^2) is the solar energy, TempOut(ºF) is the outdoor temperature, HumOut(%) is the outdoor humidity. These weather variables are from UMM campus website.

Our initial data set contained 31 variables. Variables in the final model were selected using backward selection where variables with the highest p-value are removed one-by-one in a stepwise fashion. At each step the model with the lowest AIC (Akaike Information criterion) is retained. AIC balances predictability against the number of parameters and is a common way to choose one model over another.

$$AIC = 2k + n\ln\left(\frac{RSS}{n}\right)$$

We also performed another round of backward selection using an ANOVA F test to compare the models. After these calculations, we had produced two linear models as candidates for the final model. All linear models have an associated R^2 that measures how good a job the linear model does at fitting the data. Residual standard error is another way to access strength of fit is to consider how far off the model is from perfectly fitting the data. In our case one model was superior in both respects and we picked the one with the higher R^2 and lower residual standard error to produce the final model.

Our model can be improved from the following two aspects:

1. The model may unbiased if we try more sophisticated models such as power transformations and general linear model. The residual plot of our currently model looks like a roughly linear decreasing line, which implied that the model we had is biased and homoscedastic, so we need to try more models to decrease the biase.
2. Our model is affected mostly by the variable *solar energy,* but the problem is that variable is, currently, historical data. In order for the model to be useful for predicting solar energy utilization in advance, the variable needs to be replaced with a predicted solar energy. So that will be great if we can get the directly weather data, like cloud cover and sunshine duration, and based on them to build a model for the production of solar panels. Because the predicted weather data can be reached easier than solar energy. (We are working on this improvement by collecting historical and forecasting weather data through Dark Sky API see section 2.2)

Another point of concern is whether or not our model, produced using data from the year 2015, will work well for making predictions. The year 2015 was chosen because of availability of historical data, we are not yet sure if the model that we created can explain the solar panel production of most years or make accurate predictions for future years . So we are going to verify our model by selecting another years' data.

On our future plan, we will make a model for seasonally horizontal comparison. What we have learned from our models so far all depend on the weather data and energy data for each year, and it is better for us to think about how solar panel production will change when give different years but with same season or same month. Since the sunshine duration and expected solar energy are regular for each season in a year, we can give a basic production forecast in a long run.

2.2 Using Dark Sky API for weather prediction and hence power prediction

Dark Sky is one of the many APIs online available for weather information. The Dark Sky API allows you to look up the weather anywhere on the globe, returning [3]: Current weather conditions, Minute-by-minute forecasts out to one hour, Hour-by-hour and day-by-day forecasts out to seven days and lastly Hour-by-hour and day-by-day observations going back decades. We are trying to utilize this API data from Dark Sky to incorporate on our models to predict the power production. For instance, like the model suggested above, utilizing the TempOut and HumOut prediction data gathered from the API from Dark Sky for 2018 May in Morris, Minnesota we can deduce GenSP for green prairie solar panel.

We can also get real time data for weather using Dark Sky and utilize it to model and compare with real time consumption data to analyze and predict energy prediction.

Forecast Requests can be made to Dark Sky using the format:
https://api.darksky.net/forecast/[key]/[latitude],[longitude]] Where, key is an individual access code given to a person trying to access weather prediction for the given latitude,longitude. An example of a Forecast Request for UMM would would be:
// https://api.darksky.net/forecast/*KEY*/46.7296,94.6859

2.3 Using real time consumption API to monitor and predict power consumption:

We used API(Application Programming Interface) to get energy consumption data. Since the University of Minnesota- Morris(UMM) doesn't have a database to store all electric meter, UMM sends all the data to the twin-cities campus. They store all electric meter data in their database every 5 minutes.

To predict UMM's energy consumption, we had to know UMM's historical and present/current energy consumption. For that matter by using API we were able to get energy consumption data of all buildings. The api has 4 parameters. The first one is and ID. it is a unique name/number to each building's meter. By providing the ID of a specific building on th API we can get its electric usage. The second one is start time: The date/time to start grabbing data. The third one is end time: the data/time to stop grabbing data. The last one is frequency Min which is mostly used to reduce load on the server and increase processing time. By default we can get 60 minutes(an hour) data.

For example, *em-cof.energy.umn.edu/meterapi/api/modbus/getvalues?meterId=769-EL-R01&meterId=769-EL-R101&startTime=2018-01-03&endTime=2018-01-20.* This gets values for meters *769-EL-R01 and 769-EL-R101,* which are the two electric meter Id located in Green Prairie and it gets the data from 1/3/2018 to 1/20/2018 with a default frequency of 60 minute. We were also able to use the API to plot a real time power consumption data by directly grabbing consumption data from the Twin-cities campus database.

3. Visualisations and User Interface Design (Under Development)

For the front end development of the software we will start by building a webApp that will be utilizing Angular 5, along with Ionic 3 through which we will also make it available for mobile to be downloaded as apps on iOS and android.

AngularJS 5 is one of the industry standard for webApps. A primary reasons for using AngularJS 5 for our webApp is for the UI-UX experience. The Material Design components compatible with server-side rendering is no less than a reward for users. AngularJS 5 tends to reduce loading time as well which could be crucial as our app would be handling a lot of data and charts/graphs. More loading time means more user dissatisfaction which, in turn, translates into higher app abandonment, and more dissatisfied customers or poor productivity. **[5]**

Ionic is a complete open-source SDK for hybrid mobile app development. The more recent releases, known as Ionic 3 or simply "Ionic", are built on Angular. Ionic provides tools and services for developing hybrid mobile apps using Web technologies like CSS, HTML5, and Sass. **[6]** Having a mobile app version rather than going to the web app every time would be better for students utilizing the app to plan ahead their power consuming activities. An app means we could also keep the software running in the background and send notifications to people to notify them about the specific peak hours to promote these power intensive activities.

A great deal of the front development of the software heavily relies on its implementations by the campus students to fulfil its goals. Students will be able to access via browsers on computers through a website.

We will also have individual webfiles that can be run on big screens on campus which will project real time data of consumption of the campus along with the graph of the demand being met by renewable energy sources. Some of the big screens would also be utilizing the prediction parts of the software once they are up and running.

3.1 Libraries used for charting

For the charting of the graphs, we would be utilizing Google Charts. Reasons for using Google Charts as a charting library rather than any other JavaScript charting library is because it provides many pre-built charts like area charts, bar charts, calendar charts, pie Charts, geo charts, and more. Google charts also comes with various customization options that help in changing the look of the graph. Charts are rendered using HTML5/SVG to provide cross-browser compatibility and cross platform portability to iPhones, iPads, and Android. **[7].**

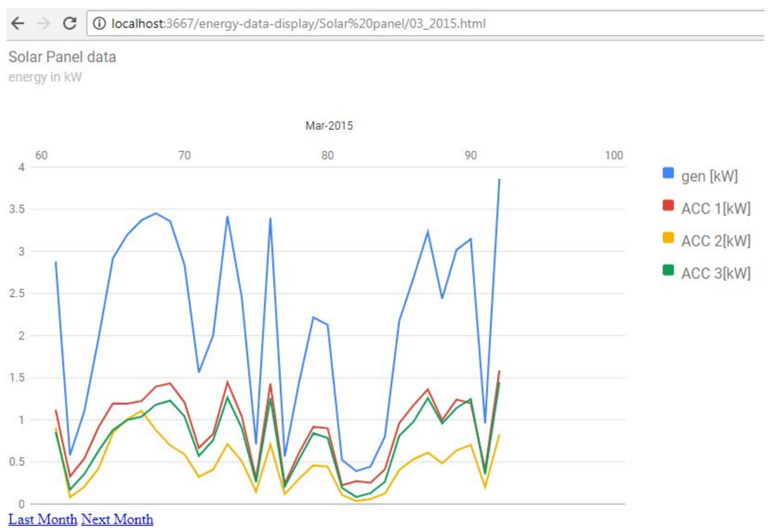

Figure:0- UMM solar panel production for 2015 January in Google Charts

3.2 User Interface for the app.

The login page is going to synced with the U of M login page. This also allows us to the login interface of the U of M rather than setting up our own list of accounts. This will also allow us to gather data later about the percentage of students of the morris campus (as every student from the campus has the same @morris.umn.edu email) utilizing the app. Feedbacks will be better summarized and more personalized. People without their morris accounts can access the general account which will named after the school mascot “Pounce, the cougar”.

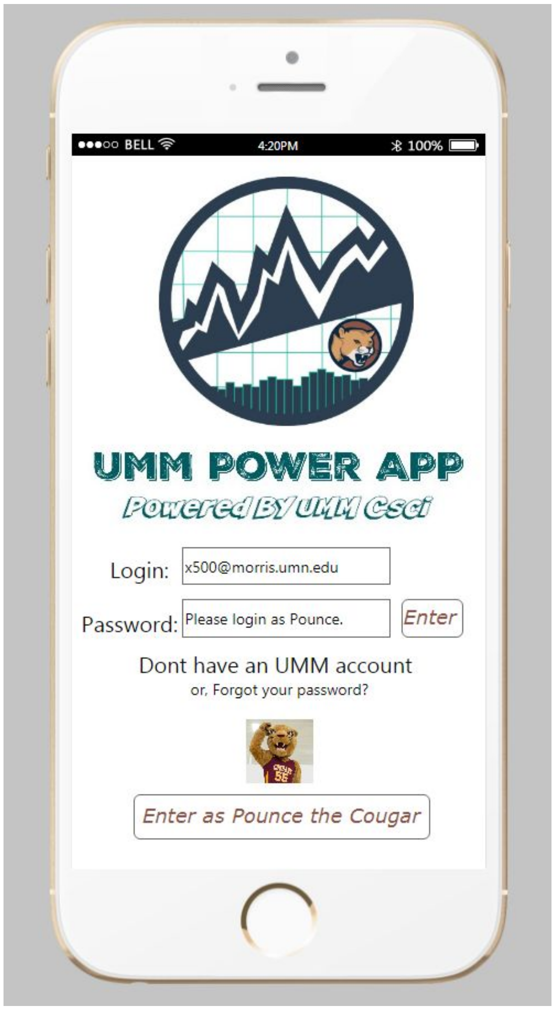

Figure:1- Login

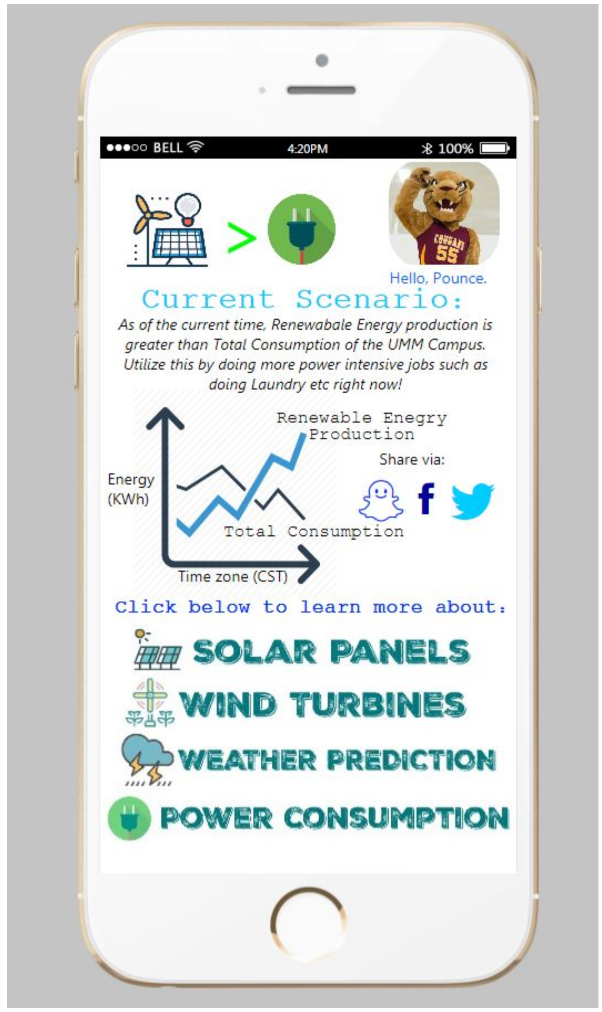

Figure:2- HomePage

The homepage will consist of the current scenario of the power consumption of the entire campus being compared to the real time renewable energy harvested with a chart being projected just underneath to show the prediction graphs of when renewable energy might be more than the power demands of the campus.

Additional pages would include : Solar Panels that tells the user more details about the location of all of the individual solar panels and just individual power production from the panels (both historic and real time), Wind Turbines that tells the user more details about the location the solar panels and just individual power production from the wind turbines (both historic and real time), Weather Prediction that will just use the DarkSky API to project weather for the morris location and lastly Power Consumption that will include historical graphs of the last years in kwH for the campus (to reflect power consumption and encourage reduction) and real time power consumption data from the API from the Twin cities campus.

We going to be single webfiles as well that will be running on big screens around campus tv screen that will feature live feed of real time data to give students an estimate of how much of the total power consumption is actually being covered by renewable energy. The charting will be done by GoogleCharts from the API real time data fields.

One thing that we are going to make sure is that since it's going to be on display for the entire campus students, that it does not ever display an empty feed. We will make sure that the entire live chart is running on a loop and on that specific occasion when server is down, the last live picture on the loop is still being displayed. For instance, if somehow the server goes down at 4pm and it's still not back and it's 8:30 PM, the display is still showing the live charts from 4pm rather than giving a server down message.

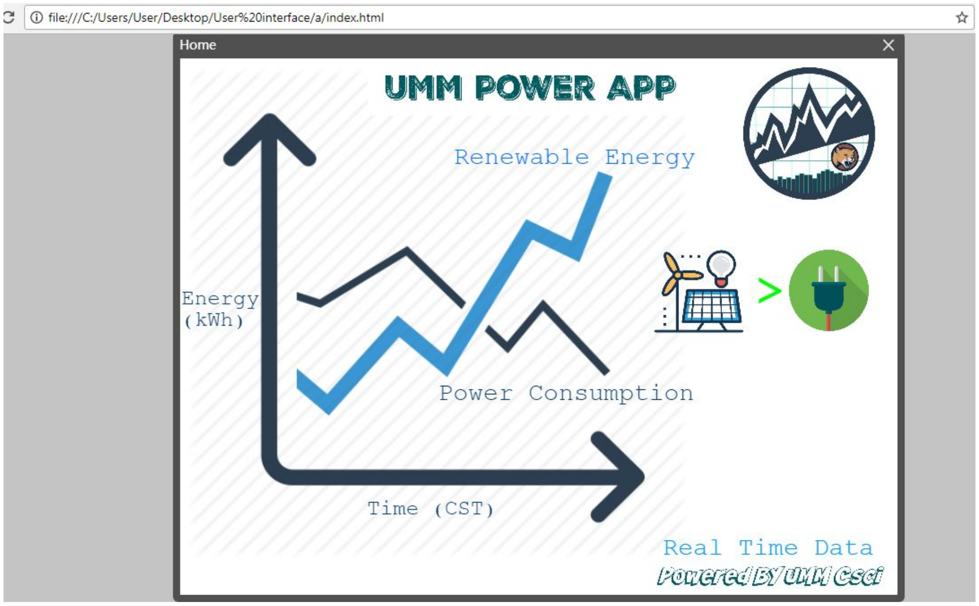

Figure 3- Single Webpages to be used in big screens

4. Purpose

The first and the main purpose of this project is to save money by fully utilizing our energy production, instead of sending the produced energy back to the grid. As we mentioned above when the turbines and the solar turbines produce more energy than the amount of energy that is needed for the buildings, we sent back the produced energy and sell it at a cheap price. But if we know when our green energy production exceeds the amount of energy needed, we can do things, like laundry that require huge power demands during those times which would potentially save a huge amount of money.

The other thing this project contributes is, to assist the school in budgeting money to spend on energy conception. Since the amount of energy that UMM demands from Ottertail varies from month to month, depending upon how much renewable energy we produce from the turbines and the solar panels, this project can help budgeting. For example, on sunny and windy months we generally produce a higher amount of energy from the turbines and the solar panels that means we buy a fewer amount of energy from Ottertail. This project helps the management to predict how much UMM pays for electric bill every week.

The detailed analysis of real time power consumption of the campus would give us insights to truly understand the nature and tweak it accordingly to our renewable sources to make sure we can maximum affordability and benefits. We can look at if having actual battery storages would be beneficial for the campus during those specific renewable peak hours. We can go beyond and try to calculate how much should we be spending on battery storages to make it a better economic alternative than to send the electricity back to the grid.

Summing up, the paper strives to prove the importance of such a software system for the UMM campus and how the system would be built with its specific needs in mind. Undoubtedly, this would a huge step towards the campuses step towards a sustainable and renewable future.

References

[1] https://weather-data.morris.umn.edu/

[2] http://egauge13797.egauge.es/

[3] Dark Sky Api - https://darksky.net/dev/docs#data-sources

[4] Angular - https://angular.io/docs

[5] Ionic - https://ionicframework.com/docs/

[6] Google Charts - https://developers.google.com/chart/interactive/docs/gallery