UNITED STATES **PROVISIONAL** PATENT APPLICATION

FOR

**METHOD AND SYSTEM FOR TESTING SYSTEM HEALTH OF A SOLAR PANEL SYSTEM**

INVENTORS:

Anika Misra (LOS ANGELES CA)

Prepared by:

Customer Number 13883

**METHOD AND SYSTEM FOR TESTING SYSTEM HEALTH OF A SOLAR PANEL SYSTEM**

**BACKGROUND**

1. As of now, companies that manage solar panel systems for homeowners and commercial properties can only reactively determine when the system is “online” and when it is “offline”. An offline status is usually provided when the inverter is not receiving any generation data, whether it be through a connectivity issue or an API issue. An online status is when generation data is being received. However, there is no way to proactively determine if the system is about to go offline. What this process does is predict if this solar panel system will go offline - hence producing a new state: “Unhealthy”, or, “Warning”.

**BRIEF DESCRIPTION OF THE DRAWINGS**

1. **Figure 1** illustrates an example process for testing system health of a solar panel system, according to some embodiments.
2. **Figures 2 A-D** illustrates another example process for testing system health of a solar panel system, according to some embodiments.
3. **Figure 3** illustrate another example process for solar panel health testing and analysis, according to some embodiments.
4. **Figure 4** illustrates an example process for utilizing a decision tree to analyze solar system health, according to some embodiments.
5. The Figures described above are a representative set and are not an exhaustive with respect to embodying the invention.

**DESCRIPTION**

1. Disclosed are a system, method, and article of manufacture of utilizing statistical outliers to test for malfunctioning solar panel systems. The following description is presented to enable a person of ordinary skill in the art to make and use the various embodiments. Descriptions of specific devices, techniques, and applications are provided only as examples. Various modifications to the examples described herein can be readily apparent to those of ordinary skill in the art, and the general principles defined herein may be applied to other examples and applications without departing from the spirit and scope of the various embodiments.
2. Reference throughout this specification to "one embodiment," "an embodiment," ‘one example,’ or similar language means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the present invention. Thus, appearances of the phrases "in one embodiment," "in an embodiment," and similar language throughout this specification may, but do not necessarily, all refer to the same embodiment.
3. Furthermore, the described features, structures, or characteristics of the invention may be combined in any suitable manner in one or more embodiments. In the following description, numerous specific details are provided, such as examples of programming, software modules, user selections, network transactions, database queries, database structures, hardware modules, hardware circuits, hardware chips, etc., to provide a thorough understanding of embodiments of the invention. One skilled in the relevant art can recognize, however, that the invention may be practiced without one or more of the specific details, or with other methods, components, materials, and so forth. In other instances, well-known structures, materials, or operations are not shown or described in detail to avoid obscuring aspects of the invention.
4. The schematic flow chart diagrams included herein are generally set forth as logical flow chart diagrams. As such, the depicted order and labeled steps are indicative of one embodiment of the presented method. Other steps and methods may be conceived that are equivalent in function, logic, or effect to one or more steps, or portions thereof, of the illustrated method. Additionally, the format and symbols employed are provided to explain the logical steps of the method and are understood not to limit the scope of the method. Although various arrow types and line types may be employed in the flow chart diagrams, and they are understood not to limit the scope of the corresponding method. Indeed, some arrows or other connectors may be used to indicate only the logical flow of the method. For instance, an arrow may indicate a waiting or monitoring period of unspecified duration between enumerated steps of the depicted method. Additionally, the order in which a particular method occurs may or may not strictly adhere to the order of the corresponding steps shown.
5. DEFINITIONS
6. Example definitions for some embodiments are now provided.
7. Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of computer programs that can teach themselves to grow and change when exposed to new data. Example machine learning techniques that can be used herein include, inter alia: decision tree learning, association rule learning, artificial neural networks, inductive logic programming, support vector machines, clustering, Bayesian networks, reinforcement learning, representation learning, similarity and metric learning, and/or sparse dictionary learning.
8. Solar power is the conversion of energy from sunlight into electricity, either directly using photovoltaics (PV) or indirectly using concentrated solar power. Photovoltaic cells convert light into an electric current using the photovoltaic effect. Solar power plants use one of two technologies: Photovoltaic (PV) systems use solar panels, either on rooftops or in ground-mounted solar farms, converting sunlight directly into electric power. Concentrated solar power (CSP) uses mirrors or lenses to concentrate sunlight to extreme heat to eventually make steam, which is converted into electricity by a turbine. A solar tracker is a device that orients a payload toward the Sun. Payloads can be, inter alia: solar panels, parabolic troughs, fresnel reflectors, lenses, or the mirrors of a heliostat.
9. Time interval can be a consistent time period where solar panel energy generation is measured such as a month, a day, a week, etc. (e.g. months can be effective, etc.).
10. EXAMPLE METHODS OF TESTING SYSTEM HEALTH OF A SOLAR PANEL SYSTEMS
11. **Figure 1** illustrates an example process 100 for testing system health of a solar panel system, according to some embodiments. Process 100 This process checks if the solar panel system (“system” for short) is going to fail. A system is considered “failing” when it records 0 kWh generation consistently for three or more days. Failure can also mean that some of the inverters have stopped working. This process uses a unique combination of checking for behavioral attributes of the system, comparing with neighboring systems, checking if error codes were received, looking at the number of inverters in the system, and more.
12. It is noted that when a solar panel system is recording 0 generation, this can mean two things. The system is actually not producing any energy (a true 0 generation). There is a connectivity issue. The system is producing energy, but this is not being communicated to the homeowner’s portal. The process outlined in this patent tells homeowners two things. If their system will fail. If this failure is true 0-generation or not. This invention is unique because it can be automated in code, therefore allowing the homeowner to not rely on customer support personnel to diagnose issues.
13. More specifically, in step 102, process 100 checks for any of the n-number health symptoms. In one example, these can be five (5) health systems discussed herein. In step 104, process 100 can ask additional physical questions. These can include, inter alia: seeing how many microinverters the system has, comparing the system performance to neighboring systems, considering recent weather, etc. In step 106, process 100 can check grid consumption as necessary. This step differentiates between a true failure and a connectivity issue.
14. **Figure 2** illustrates another example process 200 for testing system health of a solar panel system, according to some embodiments. It is noted that all health symptoms must be checked for, and it is possible to have more than one health issue. Process 200 shows an example of five health symptoms that only have “yes” as an option and continue onto the other health symptoms regardless of whether “yes” is true.
15. Definitions of common words used in process 200 are now provided:
16. ● “non-comm”: short for non-communicative, and it means that the solar panel system is not properly communicating and not showing how much energy it is producing;
17. ● “site”: this is short for homeowner site, or solar panel system;
18. ● “0-gen” or “0-generation”: short for 0 generation, and this means 0 kWh are being recorded;
19. ● “expected generation”: process 200 currently predicts how much a solar panel system should be generating for each unique site per month. These generation estimates are unique for each home: they depend on shading, panel orientation, number of panels, weather predictions, etc. Another way to calculate the expected generation is to replace “weather prediction” with “what the weather actually was” when looking at generation estimates for the previous day. This is possible to do when looking at generation on a daily level, as usually done with the Actual vs. Estimated graph. In process 200, “expected generation” can refer to two calculations. Process 200 can provide a monthly weather prediction, or, if looking at the daily, past twenty-four (24) hours level, then the same as the former but readjusted for what the weather actually was that day; and
20. ● “<” and “>”: In this flowchart, the less than and greater than symbols are not strict. They are synonymous with “less than or equal to” and “greater than or equal to”, respectively. In other words, “<” is synonymous with “≤”, and “>” is synonymous with “≥”. Now I will explain each of the health symptoms in more detail.
21. Drastic flickering change in Actual vs. Estimated graph” is now discussed. The “Actual vs. Estimated graph” is a graph that compares actual solar panel generation to estimated solar panel generation over time at a daily level. When analyzing this graph over several months, there can be a normal amount of “flickering” that changes gradually over seasons. For example, on one day, there might be 50% less generation than expected, and on the next day, there might be 50% more generation than expected. This type of deviation typically changes from ±100% to ±25% gradually throughout the seasons. However, if the deviation is high but then suddenly drops to a low number, and then stays at this low number for at least three weeks, then this is considered a drastic flickering change. Another example of a drastic flickering change is if the deviation is ±x%, but then suddenly begins only varying between 0% and -x%, and it deviates in this way for at least 3 weeks. If this health symptom occurs, the next step is to check if this homeowner site has more than one inverter or microinverters. If so, then this might be an issue that one or more of the microinverters is failing. Finally, process 200 checks if this is a true failure by checking if grid consumption has increased. If grid consumption has increased, this means the homeowner is using more energy than usual, so their system is not producing enough energy. If grid consumption has not increased, then it is diagnosed as a connectivity issue. 2. “<50% expected generation for more than one month”.
22. If a site’s expected generation falls to 50% or under on average, for an entire month, then this health symptom will be flagged. As shown, the next step is for process 200 to check if the site has multiple inverters, or micro-inverters because if so, this is a sign that one or more of them have failed. Finally, process 200 checks if it is a true failure by checking if grid consumption has increased.
23. This is when a site records 0 kWh consistently for at least 48 hours. As shown, the next step is for process 200 to check if other sites with the same type of inverter, or, if other neighboring sites also have been recording 0 generation on those same dates. If so, then this is a data reporting issue. Otherwise, As shown, the next step is for process 200 to check if grid consumption has increased to differentiate between a true inverter failure or a connectivity issue. 4. “< 80% expected generation for >5 days”.
24. Process 200 checks if the system has been producing less than or equal to 80% of its expected generation for five or more days, consistently. If so, then the next step is to check if this change is gradual. “Gradual” is defined as such: if the percentage changes by more than 30% in just one day and then stays at this low value, then this is not gradual. If not gradual, then the next step is to check if there has been a recent weather event that may have caused this. For example, snowfall may cause such a drastic change because it covers the solar panels immediately. However, if there is no drastic weather event, then process 200 can go back to health symptom #1 and continue from there. Now, if the change was gradual, then it might be a soiling or shading issue. These issues occur over time. So, if this is the case, the final step is to check grid consumption to differentiate between a non-comm issue or an actual soiling/shading issue.
25. “Skipped date” is now discussed. A skipped date can occur there when no information has been recorded in the API, not even a date. This can be different from a 0-gen, because with a 0-gen, “0 kWh” is recorded. In the case of a “skipped date”, the entire day is skipped in the energy production log. This is diagnosed as a data reporting issue.
26. Error code received checkpoint can be: “>1 of the above 6 health symptoms observed.”
27. It is noted that in process 200, the decision tree on the health symptoms is not binary. If “yes” is selected for one health symptom, then other symptoms must still be checked for. This is because it is possible to have more than one health symptom. Process 200 ensures that all health symptoms will be checked for.
28. The purple diamond step of process 200 serves as a checkpoint to see if at least one health symptom has been flagged. If no health symptoms have been flagged, then the last step in the system health process is to check for any error codes received from the homeowner’s solar panel portal. If there are no error codes received, then process 200 can ensure that the system is truly healthy. If none of these health symptoms were flagged but an error code was still received, then there is some other system issue that is based on the error code.
29. If health symptoms have been flagged, it is still necessary to check for error codes. If an error code was received, the diagnosis should be based on the error code received along with what was diagnosed on the flow chart. If no error code was received, then the system is still unhealthy, based on the flow chart health symptom path. This marks the end of the explanation of the flow chart. **Appendix A** includes a wholistic view of process 200.
30. **Figure 3** illustrate another example process 300 for solar panel health testing and analysis, according to some embodiments. In step 302, process 300 creates a new system feature for solar panel homeowners, which is “unhealthy” or “warning”. No other solar panel management technology companies have this state. They only have online and offline.
31. In step 304, process 300 automates the troubleshooting process as opposed to relying on customer support personnel. In step 306, process 300 predicts when the inverter will record 0-generation 4. Process 300 is able to detect when something is a connectivity issue or truly a malfunctioning system.
32. In step 308, process 300 combines steps 302 and 304. In step 310, process 300 incorporates real-time weather data into the generation estimate and uses the real weather data to modify the generation estimate. No other solar panel management technology companies have this idea implemented in an automated way. In step 312, process 300 determines multiple system health issues such as, inter alia: a. malfunctioning inverter, b. Wi-Fi or connectivity issue, c. soiling, d. shading, e. inverter-wide data issue, etc.
33. Process 100-300 differ from current technology for the following reasons. No other solar panel management technology companies have an automated system to troubleshoot the most common issues of solar panel systems, such as malfunctioning inverters, connectivity issues, soiling and shading issues, and inverter-wide data issues. No other solar panel management technology companies combine the usage of error codes and health symptoms in this automated way. No other solar panel management company incorporates real weather data to modify generation estimates. Not only do process 100-300 predict when the inverter will fail, but it is able to determine if this failure is a true failure or just a connectivity issue. No other solar panel management technology company has a way to automate an “unhealthy” or “warning” state for solar panel systems. Process 100-300 can be used to diagnose installer-wide and inverter-wide data issues, not just for the individual home, as most other companies focus on. Process 100-300 enables diagnoses of multiple health symptoms at once.
34. **Figure 4** illustrates an example process 400 for utilizing a decision tree to analyze solar system health, according to some embodiments. Most decisions (e.g. in the diamonds) have two options, yes or no, that lead you to a different step in the process. However, the health symptoms have only “yes” and then continue to check for other health symptoms. This is done because all health symptoms must be checked for. A detailed description of each step is provided in the detailed description of the invention section above. One algorithm that is used is arithmetic in the “Actual vs Estimated graph”. In this case, the way this percentage is calculated is through a simple “percentage error” calculation: percentage error = 𝑎𝑐𝑡𝑢𝑎𝑙 𝑔𝑒𝑛𝑒𝑟𝑎𝑡𝑖𝑜𝑛/𝑒𝑠𝑡𝑖𝑚𝑎𝑡𝑒𝑑 𝑔𝑒𝑛𝑒𝑟𝑎𝑡𝑖𝑜𝑛 𝑒𝑠𝑡𝑖𝑚𝑎𝑡𝑒𝑑 𝑔𝑒𝑛𝑒𝑟𝑎𝑡𝑖𝑜𝑛.
35. In the “Actual vs. Estimated Graph”, in addition to the percentage error calculation, several machine learning techniques are used, along with technical systems and algorithms.
36. In step 402, process 400 can obtain the data for “estimated” generation. Process 400 can first use a specified software and model to generate a generation prediction for ideal weather conditions based on the homeowner’s site conditions such as soiling, shading, position of solar panels, number of solar panels, etc. “Weather data” includes but is not limited to temperature, graphical horizontal irradiance, direct normal irradiance, snowfall, precipitation, and more.
37. In step 404, process 400 can generate estimated generation as monthly total predictions. To obtain the daily total generation estimate, process 400 can divide each monthly total by the number of days in a month. daily generation estimate #1 = 𝑚𝑜𝑛𝑡ℎ𝑙𝑦 𝑔𝑒𝑛𝑒𝑟𝑎𝑡𝑖𝑜𝑛 𝑒𝑠𝑡𝑖𝑚𝑎𝑡𝑒/𝑛𝑢𝑚𝑏𝑒𝑟 𝑜𝑓 𝑑𝑎𝑦𝑠 𝑖𝑛 𝑠𝑎𝑖𝑑 𝑚𝑜𝑛𝑡ℎ. And process 400 can do this for every month of the year for each site. This process can be automated through coding or Excel.
38. In step 406, process 400 can, for the “Actual vs. Estimated” graph, provide a daily generation estimate (e.g. see step 402, etc.) that is accurate enough to be used in the percentage error. Most other times, however, process 400 can make the daily generation estimate even more accurate. This can use machine learning to increase/optimize accuracy.
39. In step 408, process 400 can implement a machine learning component. Process 400 can determine which weather data was assumed in the original generation estimate. If the weather data is in a daily format, then, process 400 can take monthly averages of that data, and then divide all of this by the number of days in a month to make a consistent weather prediction for every day of the month.
40. A machine learning engine can utilize machine learning algorithms to optimize the methods herein. Machine learning is a type of artificial intelligence (AI) that provides computers with the ability to learn without being explicitly programmed. Machine learning focuses on the development of computer programs that can teach themselves to grow and change when exposed to new data. Example machine learning techniques that can be used herein include, inter alia: decision tree learning, association rule learning, artificial neural networks, inductive logic programming, support vector machines, clustering, Bayesian networks, reinforcement learning, representation learning, similarity, and metric learning, and/or sparse dictionary learning. Random forests (RF) (e.g. random decision forests) are an ensemble learning method for classification, regression and other tasks, which operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (e.g. classification) or mean prediction (e.g. regression) of the individual trees. RFs can correct for decision trees’ habit of overfitting to their training set. Deep learning is a family of machine learning methods based on learning data representations. Learning can be supervised, semi-supervised or unsupervised.
41. Machine learning can be used to study and construct algorithms that can learn from and make predictions on data. These algorithms can work by making data-driven predictions or decisions, through building a mathematical model from input data. The data used to build the final model usually comes from multiple datasets. In particular, three data sets are commonly used in different stages of the creation of the model. The model is initially fit on a training dataset, which is a set of examples used to fit the parameters (e.g. weights of connections between neurons in artificial neural networks) of the model. The model (e.g. a neural net or a naive Bayes classifier) is trained on the training dataset using a supervised learning method (e.g. gradient descent or stochastic gradient descent). In practice, the training dataset often consist of pairs of an input vector (or scalar) and the corresponding output vector (or scalar), which is commonly denoted as the target (or label). The current model is run with the training dataset and produces a result, which is then compared with the target, for each input vector in the training dataset. Based on the result of the comparison and the specific learning algorithm being used, the parameters of the model are adjusted. The model fitting can include both variable selection and parameter estimation. Successively, the fitted model is used to predict the responses for the observations in a second dataset called the validation dataset. The validation dataset provides an unbiased evaluation of a model fit on the training dataset while tuning the model’s hyperparameters (e.g. the number of hidden units in a neural network). Validation datasets can be used for regularization by early stopping: stop training when the error on the validation dataset increases, as this is a sign of overfitting to the training dataset. This procedure is complicated in practice by the fact that the validation dataset’s error may fluctuate during training, producing multiple local minima. This complication has led to the creation of many ad-hoc rules for deciding when overfitting has truly begun. Finally, the test dataset is a dataset used to provide an unbiased evaluation of a final model fit on the training dataset. If the data in the test dataset has never been used in training (e.g. in cross-validation), the test dataset is also called a holdout dataset.
42. Returning to step 408, this step can be modified and skipped. There might not be a need to make the daily weather data consistent for each day of the month. Process 400 can use daily generation estimate of step 402.
43. Process 400 can then train any machine learning model, such as linear regression or neural networks, to create a model that associates the daily weather with the daily generation estimate of step 402. So, the input of the model should be “weather for one day”, and the output of the model should be “daily generation estimate”.
44. It is noted that this model is different for every site, since each site has different monthly generation estimates based on site conditions. After the model has been properly trained, process 400 can use real-time weather data to determine what the weather was in the past 24 hours for a site. This step can be modified to any short timeframe from 1 hour to the past week or month, based on what is most accurate.
45. Finally, process 400 can plug in the average weather from the past 24 hours (e.g. or any given timeframe) into our model and the output should be the expected generation based on this weather input. This is known as a daily generation estimate. This can be modified to weekly or monthly generation estimate. The main purpose of the machine learning step is to incorporate real weather data into the generation estimate, as no other solar panel management companies do this.
46. In step 410, process 400 provides a generation estimate based on what the weather was that day, process 400 can look at what the site’s solar panel system actually generated, and process 400 can plug these values in the percentage error: percentage error = (𝑎𝑐𝑡𝑢𝑎𝑙 𝑑𝑎𝑖𝑙𝑦 𝑔𝑒𝑛𝑒𝑟𝑎𝑡𝑖𝑜𝑛 − 𝑑𝑎𝑖𝑙𝑦 𝑔𝑒𝑛𝑒𝑟𝑎𝑡𝑖𝑜𝑛 𝑒𝑠𝑡𝑖𝑚𝑎𝑡𝑒)/𝑑𝑎𝑖𝑙𝑦 𝑔𝑒𝑛𝑒𝑟𝑎𝑡𝑖𝑜𝑛 𝑒𝑠𝑡𝑖𝑚𝑎𝑡𝑒.
47. In step 412, process 400 plots these values for each day of the year and then look at trends based on the plot. For example, as shown in the flowchart, if there is less than 80% than expected generation for five (5) or more days, then the system issue might be snowfall.
48. It is noted that sometimes when weather data is not available, or process 400 may look at general trends quickly, accordingly, process 400 can use daily generation estimate in the percentage error in step 414.
49. CONCLUSION
50. Although the present embodiments have been described with reference to specific example embodiments, various modifications and changes can be made to these embodiments without departing from the broader spirit and scope of the various embodiments. For example, the various devices, modules, etc. described herein can be enabled and operated using hardware circuitry, firmware, software or any combination of hardware, firmware, and software (e.g., embodied in a machine-readable medium).
51. In addition, it can be appreciated that the various operations, processes, and methods disclosed herein can be embodied in a machine-readable medium and/or a machine accessible medium compatible with a data processing system (e.g., a computer system), and can be performed in any order (e.g., including using means for achieving the various operations). Accordingly, the specification and drawings are to be regarded in an illustrative rather than a restrictive sense. In some embodiments, the machine-readable medium can be a non-transitory form of machine-readable medium.