

Solarenergie

JONAS BUCHBERGER, Technische Universität München

Die Nutzung der Solarenergie bringt einige Probleme mit sich. Vor allem die genaue Vorhersage ist in der Energieproduktion von Wichtigkeit. In diesem Artikel wird näher auf einige Probleme eingegangen und ein Lösungsansatz für die Vorhersage von Solarenergie dargestellt.

ACM Reference format:

Jonas Buchberger. 2017. Solarenergie. 1, 1, Article 1 (July 2017), 6 pages.
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

CONTENTS

Abstract	1
Contents	1
1 Einleitung	1
2 Verwendung von Solarenergie	2
3 Probleme	2
4 Lösungsansatz	2
4.1 Allgemein	2
4.2 Genaue Umsetzung	2
4.3 Probleme der Umsetzung	3
4.4 Verbesserungen der Umsetzung	3
5 Herausforderungen	3
6 Credits	4
References	6

1 EINLEITUNG

Die Globaleerwärmung steht vor der Tür. Das ist sicher. Obwohl es viele "Climate Change Gegner" gibt, ist der Sachverhalt deutlich. Durch die exzessive Nutzung fossiler Brennstoffe reichert sich die Atmosphäre der Erde mit Kohlenstoffdioxid (CO²) an. Durch den gesteigerten Kohlenstoffdioxidgehalt werden Sonnenstrahlen öfter zwischen Erdoberfläche und Atmosphäre reflektiert als normal. Es entsteht der so genannte Treibhauseffekt, der für die Globaleerwärmung verantwortlich ist. Um den Kohlenstoffdioxidausstoß zu verringern, wird die Nutzung von erneuerbaren Energien von immer größer Bedeutung und stellt die Forscher vor große Herausforderungen[6].

2 VERWENDUNG VON SOLARENERGIE

Derzeit ist der Anteil der Energiegewinnung durch Solarenergie in Deutschland sehr gering. Das liegt daran, dass hauptsächlich Photovoltaikanlagen oder thermische Solaranlagen, nur im Eigenheim oder im Gewerbe verwendet werden und dort oft nicht für die komplette Energieversorgung tagsüber reicht. Große Solarparks, wie es einige in Indien und den USA gibt, sind in Deutschland bis jetzt nicht zu finden. Das hat auch seine Gründe, auf welche im Folgenden eingegangen werden[1].

3 PROBLEME

Die Nutzung von Solarenergie bringt viele Probleme mit sich. Dazu gehört zum einen die sehr geringe Energiedichte der Photovoltaik-Anlagen. Man benötigt riesige Flächen für Solarparks, damit diese mit den modernen Kohle- oder Atomkraftwerken konkurrieren können. Zudem kommt noch die hohen Anschaffungs- und Wartungskosten. Damit Solarparks rentabel sind, müssen sie im Schnitt 10 Jahre ohne große Probleme Stromerzeugen. Dabei sollten täglich die Solarmodule gereinigt werden um eine maximale Energieausbeute zu sichern. Ein weiteres Problem ist die Speicherung der erzeugten Energie. Derzeit besteht keine Möglichkeit eine sehr große Menge an Energie zu speichern. Somit ist es nicht möglich, zum Beispiel eine Stadt, ausschließlich mit Solarenergie zu versorgen, da diese auch in der Nacht mit Strom benötigt. Was uns auch schon direkt zum größten Problem führt. Solarenergie ist witterungsabhängig. Die Erzeugung von Energie mit der Sonne hängt von vielen Faktoren ab, wie zum Beispiel Regen, Sonnenstunden, Luftfeuchtigkeit und Bewölkungsdichte[1].

4 LÖSUNGSANSATZ

4.1 Allgemein

Um den Problem der Witterungsabhängigkeit entgegen zu wirken, werden vor allem sehr genaue Wetterprognosen benötigt. Diese können von Wetterstationen bereitgestellt werden. Anhand dieser Daten kann dann die solare Energie, welche an einem bestimmten Ort auf die Erdoberfläche strahlt, bestimmt werden. Somit können langjährige Energieversorgungspläne erstellt werden, um eine sichere Energieversorgung zu gewährleisten.

Aber wie bestimmt man aus den Wetterdaten die nutzbare Sonnenenergie? Anhand von Energiedaten aus vergangenen Jahren und den dazugehörigen Wetterdaten, ist es mit Machine Learning und mit Hilfe neuer Wetterdaten möglich, die Energiewerte vorhersagen.

4.2 Genaue Umsetzung

Dieser Lösungsansatz kann auch in dem Wettbewerb "AMS 2013-2014 Solar Energy Prediction Contest" verfolgt werden[5]. Dieser Wettbewerb wurde von der AMS (American Meteorological Society) veranstaltet und über die Website Kaggle veröffentlicht. Kaggle ist eine weitverbreitete Plattform, die eine Vielzahl an AI Wettbewerben bietet[3]. Auch wenn dieser Wettbewerb schon beendet ist, stehen die gesamten Wetterdaten weiterhin zu Verfügung. Genau geht es in diesem Wettbewerb darum, dass man mit Hilfe der GEFS-Wetterdaten (Global Ensemble Forecast System) und den Solarenergie-Daten der Mesonet Stationen die erwartete Solarenergie der nächsten Jahre in Oklahoma bestimmt. Zu sehen sind die Stationen auf dem Bild 1 auf Seite 4.

Die zu Verfügung gestellten netCDF4 Wetterdaten erstreckten sich im Zeitraum von 1994 bis einschließlich 2007. Sie beinhalten 14 Features, wie maximale und minimale Temperatur, Luftdruck, Lang- und Kurzwellenstrahlung, Feuchtigkeit usw. Alle Werte wurden von 12:00 Uhr bis 24:00 Uhr im 3-Stunden-Takt gemessen. Dabei wurde jeder

Wert mit 11 verschiedenen Methoden aufgenommen. Zusätzlich sind noch die Längen und Breitengrade zu jedem Wert geben. Die Solarenergie Werte in Jouls pro m^2 befinden sich in einer zusätzlichen CSV Datei, zu jedem der Mesonet Stationen.

Die jetzige Version des Scripts, kann für gegebene Längen- und Breitengrade und der zugehörigen Stationnummer, die Solarenergie in Jouls pro m^2 für den Zeitraum von 2005 bis einschließlich 2007 vorherzusagen.

Dafür wurden zuerst alle netCDF4 Dateien zusammen gefasst, um sie weiter verwenden zu können. Als erstes wird der Durchschnitt über alle 11 Aufnahmemodelle und über die Tageswerte, so dass pro Tag nur ein Wert für die 14 Features gegeben ist. Das gleiche macht man dann für die restlichen 13 Dateien und fügt alle an einander in eine CSV Datei ein. So erhält man eine zwei-dimensionale Matrix, die man ohne Probleme von einer SVM lesen kann. Einen kurzen Ausschnitt sieht man in Bild 2 auf Seite 4, wobei jede Spalte für eines der 14 Features steht. Um Vergleichswerte zu haben wurde der Trainings-Datensatz zerteilt, damit man später das Ergebnis mit den realen Werten vergleichen kann. So werden die ersten 11 Jahre zum trainieren verwendet und die restlichen 3 Jahre zum Vergleich. Da es sich bei dieser Art von Problemstellung um nicht linearer Regression handelt, kann mit Hilfe eines Kernel-Ridge-Regressors die Solarenergie vorher gesagt werden[7]. Vorerst wird noch nicht der Höhen- und Lageunterschied der GEFS und Mesonet Stationen beachtet, deshalb ist eine sinnvolle Aussage nur möglich bei Stationen die sehr nah bei einander liegen.

Die erhaltenen Werte für Tahlequah sieht man im Diagramm 3 auf Seite 5. Was uns auch direkt zu den Problemen der Implementierung bringt.

4.3 Probleme der Umsetzung

Wie man auf dem Graphen 4 auf Seite 5 sehr gut erkennen kann, ist der unterschied zwischen den vorhergesagten und den gemessenen Werten manchmal sehr groß. Die Differenz liegt im Durchschnitt bei ungefähr 500000 Jouls pro m^2 . Das ist für eine genaue Erstellung eines Energieversorgungsplans definitiv zu hoch. Diese Ungenauigkeiten könnten mehrere Gründe haben:

- (1) Vereinfachung der Wetterdaten.
- (2) Höhenunterschiede der Stationen nicht beachtet.
- (3) Distanzunterschiede zwischen den Stationen nicht beachtet.
- (4) Datensatz zum Trainieren des Regressors zu klein.
- (5) MAE (mittlerer absoluter Fehler) nicht verwendet.

4.4 Verbesserungen der Umsetzung

Um (4) zu Verbessern könnte man zusätzlich die Daten der umliegenden Stationen zum trainieren des Regressors verwenden. Dadurch hätte man einen viel größeren Datensatz und genauere Ergebnisse. Zusätzlich sollte man für (1), nicht über die Stunden den Durchschnitt bilden. Dadurch geht einiges an Genauigkeit verloren. Dafür wäre der Aufbau der Trainingsmatrix viel aufwändiger. Außerdem könnte man noch eine genauere Alpha-Wert-Berechnung für den Regressor durchführen und den Höhenunterschied mit ein berechnen.

5 HERAUSFORDERUNGEN

Die größte Herausforderung war es sich in sämtliche Python Bibliotheken einzulesen und dessen Features richtig zu verwenden. Besonders die Regressoren stellten sich als nicht allzu einfach dar. Auch das einlesen in den Wettbewerb

und das verstehen der Daten nahm einige Zeit in Anspruch.

Auch wenn der vorgestellte Lösungsansatz nicht perfekt ist, zeigt er dennoch, dass es möglich ist mit so einem Verfahren genaue Energiewerte vorherzusagen. Das zeigten auch die Sieger des Wettbewerbs.

6 CREDITS

Das Programm ist in Python 2.7.13 geschrieben und verwendet die folgenden Bibliotheken:

- (1) netCDF4[8], zum Öffnen der Wetterdaten.
- (2) pandas[4] und numpy[4], zum Bearbeiten von Matrizen.
- (3) sklearn[7], für nicht linearen Regressor.
- (4) matplotlib[2], zum Plotten der Graphen.

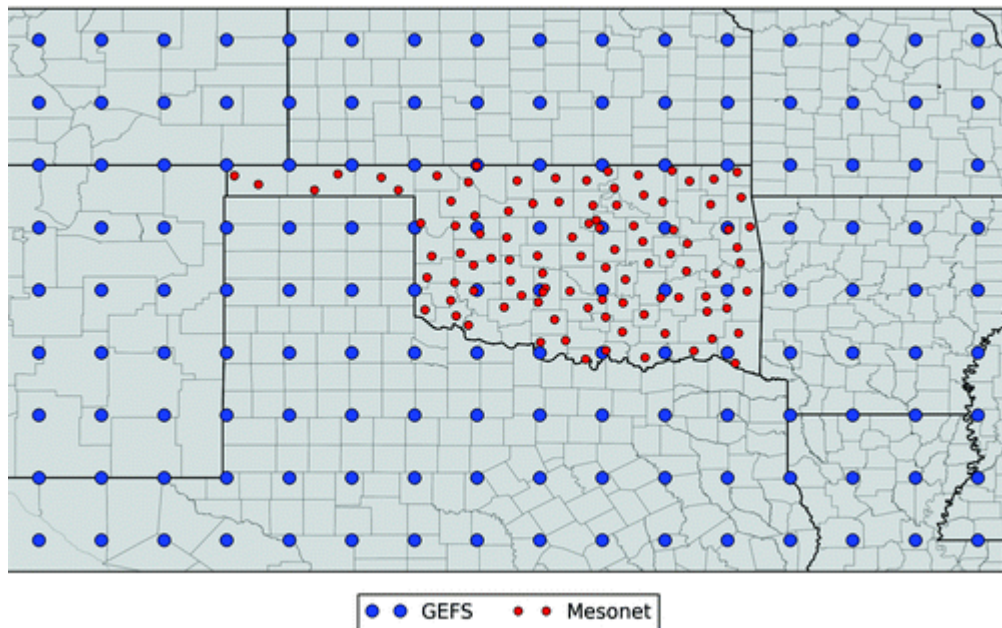


Fig. 1. Übersicht über die GEFS und Mesonet Stationen[5].

C6															
	A	B	C	D	E	F	G	H	I	J	K	L	M		
1	140.000000	346.114624	23.345455	376.574310	254.965622	19.042662	0.119636	0.937636	101907.281250	0.008137	0.119511	286.797058	283.530029	285.767029	285.797638
2	246.000000	277.798767	45.454548	366.602570	260.532715	13.129229	0.000000	0.000000	101398.257812	0.006015	0.000378	286.587219	280.283813	284.787659	284.788513
3	253.436371	246.337494	47.672729	347.113831	252.607101	6.907969	0.000727	0.000364	101817.742188	0.003152	0.001620	280.646057	277.431152	278.790466	279.239014

Fig. 2. Ausschnitt aus den zusammengefassten Daten.

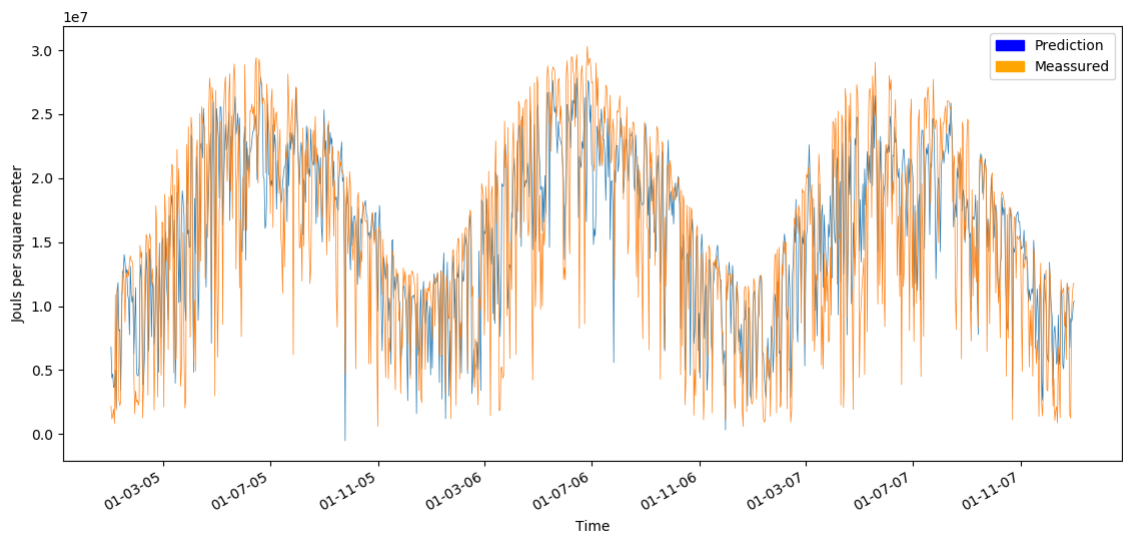


Fig. 3. Solarenergie von Tahlequah (Oklahoma) (lat: 39 lon: -95).

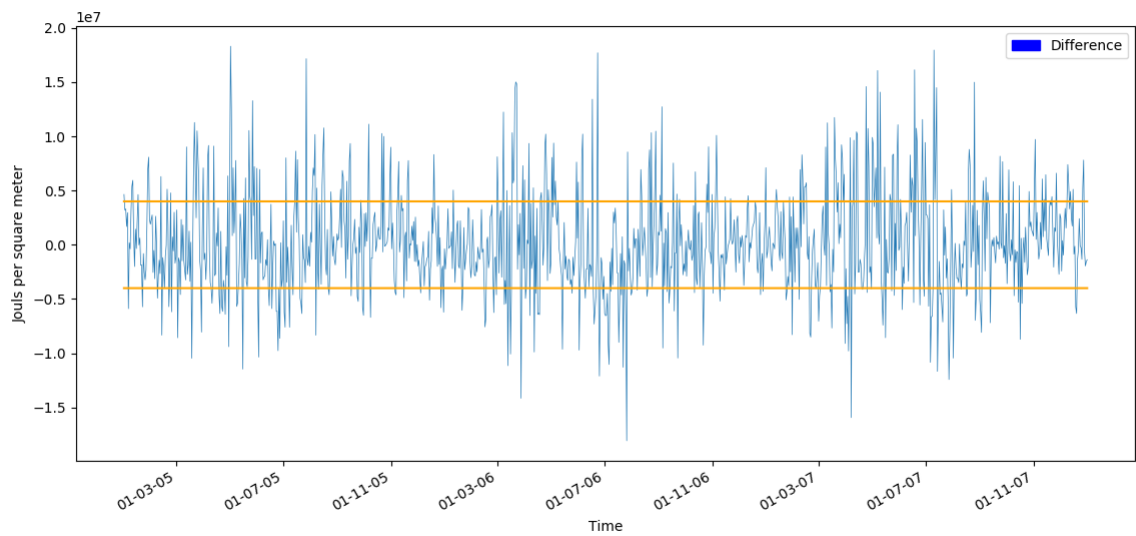


Fig. 4. Differenz der vorhergesagten und der gemessenen Solarenergie von Tahlequah (Oklahoma) (lat: 39 lon: -95).

REFERENCES

- [1] Andre Finzel. 2015. Die Solarenergie - Sonnenenergie. (2015). Retrieved July 20, 2017 from <http://www.solarenergie-sonnenenergie.com/>
- [2] J.D. Hunter. 2007. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering* 9, 3 (2007), 90–95. <https://doi.org/10.1109/MCSE.2007.55>
- [3] Kaeggle Inc. 2015. AMS 2013-2014 Solar Energy Prediction Contest. (2015). Retrieved July 06, 2017 from <https://www.kaggle.com/c/ams-2014-solar-energy-prediction-contest>
- [4] Eric Jones, Travis Oliphant, Pearu Peterson, et al. 2001–2017. SciPy: Open source scientific tools for Python. (2001–2017). Retrieved July 06, 2017 from <http://www.scipy.org/>
- [5] Amy McGovern, David John Gagne II, Jeffrey Basara, Thomas M. Hamill, and David Margolin. 2015. Solar Energy Prediction: An International Contest to Initiate Interdisciplinary Research on Compelling Meteorological Problems. *American Metrological Society* 96 (2015), 1388–1395. <https://doi.org/10.1175/BAMS-D-14-00006.1>
- [6] Die Welt Online. 2017. Und den Klimawandel gibt es doch! (2017). Retrieved July 20, 2017 from <http://www.zeit.de/wissen/umwelt/2017-05/globale-erwarmung-klimawandel-klimaforschung-klimamodelle/komplettansicht>
- [7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
- [8] Unidata. 2016. NetCDF version 4.4.1.1. (2016). Retrieved July 06, 2017 from <https://doi.org/10.5065/D6H70CW6>