

Table of Contents

- [1. task](#)
- [2. research](#)
 - [2.1. Overlook](#)
 - [2.2. 1. 2021 A Critical Review of Online Battery Remaining Useful Lifetime Prediction Methods](#)
 - [2.3. 2. 2023 Data-driven prediction of battery cycle life before capacity degradation](#)
 - [2.4. 3. 2020 Closed-loop optimization of fast-charging protocols for batteries with machine learning](#)
 - [2.5. 2022 Python - battery electro-thermal model](#)
 - [2.6. articles with open source](#)
 - [2.7. adaptive filter](#)
 - [2.8. open-source tools:](#)
- [3. Ответ rus](#)
 - [3.1. 1. Какие методы мне знакомы? Какие можно быстро применить?](#)
 - [3.2. 2. Оценить понятность методов и кода в статьях 2 и 3, оценить время адаптации подходов.](#)
 - [3.3. 3. Предложить план работ.](#)
 - [3.4. 4. Какие потребуются вычислительные ресурсы или софт.](#)
- [4. Answer eng](#)
 - [4.1. 1. What methods are familiar to me? Which ones can be applied quickly?](#)
 - [4.2. 2. To assess the clarity of the methods and code in articles 2 and 3, to assess the time of adaptation of the approaches.](#)
 - [4.3. 3. Propose a work plan.](#)
 - [4.4. 4. What computing resources or software will be required.](#)

1. task

По договоренности направляю доп. материалы для ознакомления. Они касаются подходов к прогнозированию деградации единичного аккумулятора (составляющая №1 из моего рассказа о деградации батареи; составляющую 2 - деформацию исходного распределения из-за анизотропии условий пока не рассматриваем).

Хотел бы предложить ознакомиться, а также:

1. По статье №1 ("A Critical Review of Online Battery Remaining Useful Lifetime Prediction Methods") - посмотреть, есть ли среди

методов, приведенных в обзоре, какие-то хорошо знакомые лично вам (разумеется, по другим предметным областям), с которых можно было бы начать и максимально быстро адаптировать для обсуждаемых задач.

1. Статьи 2-3 это пример метода, который я предлагаю взять для начала за основу (но все обсуждается, см. п.1). Это две статьи

одного авторского коллектива. По ним предлагаю оценить понятность метода и программного кода, который доступен по ссылкам, а также время, которое

потребуется чтобы его воспроизвести и адаптировать его для наших целей (прототип для последующей настройки/обучения/модификации и использования на собственной статистике).

1. Еще было бы хорошо попробовать подумать, какими с вашей точки зрения должны быть этапы работы (укрупненный план основных

ближайших действий) и какие входные данные с вашей точки зрения нам потребуются - потом обсудим/сопоставим наши с вами мнения.

Также, хотел бы узнать ваше мнение:

1. Потребуется ли уже на старте или в кратчайшей перспективе что-то из специальных вычислительных средств и ПО (продвинутый

компьютер, программные пакеты для обработки данных и др.)

Все это было бы хорошо обсудить, например, по телефону примерно через неделю - полторы. Будут вопросы – звоните или пишите. Мне еще хотелось бы посмотреть перечень курсов/спецкурсов, которые были у вас в ВУЗе, в идеале – в виде копии вкладыша.

2. research

2.1. Overview

Нужно сделать предсказание ресурса батареи после моделирования временного процесса деградации батареи. Модель должна обобщить процессы: химические, электрические и механические.

Идеальное решение, это модель, которая на входе имеет временные характеристики и дополнительную теоретическую информацию, а на выходе предсказание жизни батареи. Баланс между данными экспериментов и теоретическими сведениями определяет метод. Сейчас теоретические сведения статически задаются на этапе дизайна модели.

Самая простая модель, но жадная к данным - предсказатель.

Самая сложная, - это временная математическая модель, потребует решение системы уравнений. Однако конечное решение получается более простым.

Отдельный интерес представляют методы сокращения количества экспериментирований для прогнозирования.

2.2. 1. 2021 A Critical Review of Online Battery Remaining Useful Lifetime Prediction Methods

<https://www.frontiersin.org/articles/10.3389/fmech.2021.719718/full> how to select method:

- terms of input and output complexity
- percentage of the training period
- prediction accuracy.
- pros and cons of generalization

methods:

- electrochemical models - more advanced, require disassembly of cell.
- model-based - complicated
- data-driven - does not consider electrochemical reactions.

approaches

- machine-learning - require more historical data. Attractive for high-rate operating conditions.
- adaptive filter - can make an early prediction of the system state based on the confidence interval
- stochastic - bayesian estimation, Gaussian Process Regression, wiener process

Training period

- 50% for machine learning.
- 60% for adaptive filters.

Machine learning have better accuracy, but have more data, require computation time. More data and cloud computing will be helpful.

2.3. 2. 2023 Data-driven prediction of battery cycle life before capacity degradation

challenges:

- diverse aging mechanisms
- device variability
- dynamic operating conditions
- non-linear degradation process
- negligible ['negləjəb(ə)l] capacity degradation in early cycles
- small dataset of early cycles that can span degradation
- using only high-rate data is challenging because of convolution of kinetics with open-circuit behaviour

task

- quantitatively predict cycle life
- classify into two groups

Our approach is Data-based: features used in regularized linear framework, the **elastic net** (is a combination of the two most popular regularized variants of linear regression: ridge and lasso). Allow maintain interpretability. To choose hyperparameters, we apply four-fold cross-validation and Monte Carlo sampling.

Main features: voltage curves

2.4. 3. 2020 Closed-loop optimization of fast-charging protocols for batteries with machine learning

Design parameters in optimization during material selection, cell manufacturing and operation.

Methodology: Early-reduction model and Bayesian optimization algorithm, which reduces the number of experiments by balancing exploration and exploitation to efficiently probe the parameter space of charging protocols.

Optimal experimental design (OED) - typically is a closed-loop optimization system (CLO) is pipeline where feedback from completed experiments informs subsequent experimental decisions, balancing the competing demands of exploration. Applied to optimise fast-charging protocols for lithium-ion batteries.

2.5. 2022 Python - battery electro-thermal model

<https://iopscience.iop.org/article/10.1088/1742-6596/2330/1/012010>

Algorithm an iterative procedure based on the bisection method was implemented to solve the system.

2.6. articles with open source

open source

- 2020 <https://medium.com/batterybits/the-battery-software-open-source-landscape-933b88957ef5> my
 - 2023 <https://paperswithcode.com/paper/predicting-battery-lifetime-under-varying>
 - 2023 <https://github.com/microsoft/batteryml> <https://paperswithcode.com/paper/batteryml-an-open-source-platform-for-machine>
 - predicting the performance degradation of lithium batteries
 - feature extractors + models
 - PyTorch
 - 2023 Predicting Battery Lifetime Under Varying <https://arxiv.org/abs/2307.08382>
 - 2023 Lithium-ion battery degradation <https://arxiv.org/abs/2311.05482>

2.7. adaptive filter

- <https://www.intechopen.com/chapters/16112>
- https://course.ece.cmu.edu/~ece491/lectures/L27/AdaptiveFilteringChap_ADSP.pdf

goal: noise suppression or signal prediction.

types

- Conventional filters - have threshold that cut signal. inner Coefficients choosed at design step.
- Adaptive filters - coefficients are allowed to vary over time to best accomplishes a particular task.

types of adaptive filters

- analogical filters are low power consuming and fast response, offset problem
- digital filters are offset free and offer an answer of greater precision.

algorithms:

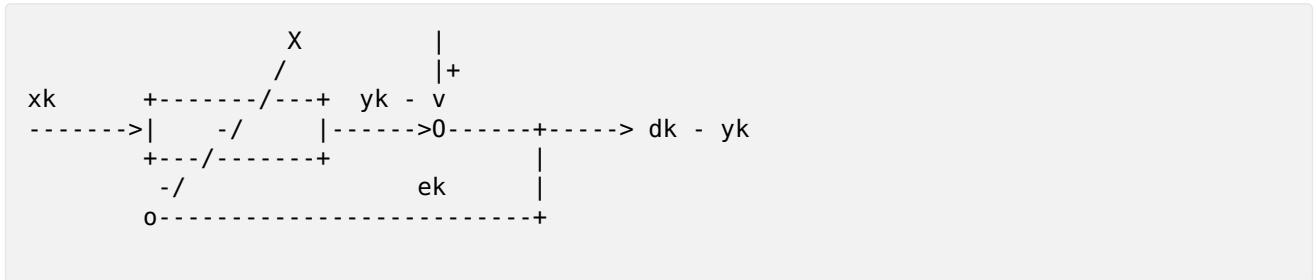
- Recursive Least Square (RLS)
- Least Mean Square (LMS),

LMS digital algorithm : $w(n+1) = w(n) + \mu e(n)x(n)$

- μ - is the filter's convergence factor.
- $x(n)$ - input signal s

hire:

- k - time
- d_k - desired signal
-



2.8. open-source tools:

- 463f <https://github.com/pybamm-team/PyBaMM>
- 97f <https://github.com/ECSHackWeek/impedance.py>
- 78f <https://github.com/microsoft/batteryml>
- 54f <https://github.com/TRI-AMDD/beep>
- 42f <https://github.com/lionsimbatoolbox/LIONSIMBA>
- 25f <https://github.com/jepegit/cellpy>

3. Ответ rus

Как видите, в статье 1 делается значительный акцент на методы основанные на данных, методы глубокого обучения. Поэтому вам нужен программист-инженер по машинному обучению.

3.1. 1. Какие методы мне знакомы? Какие можно быстро применить?

Мне хорошо знакомы глубокое обучение нейронных сетей и классические стохастические методы.

Адаптивные фильтры мне были совершенно не известны, однако это просто приложение методов регрессивного анализа, я их изучу при необходимости.

Быстро можно применить стохастические методы. Качественнее будет взять один из фреймворков и просто использовать проверенные модели, возможно предобученные, на современных архитектурах нейронных сетей, не забыв оценить наши возможности по интерпретации.

3.2. 2. Оценить понятность методов и кода в статьях 2 и 3,

оценить время адаптации подходов.

В статье 2 код очень простой, но не документирован и не организован. Чтобы изучить код потребуется около месяца. Сложность создает использование проприетарного языка MATLAB, который я не использовал много лет, и моя неквалифицированность в предметной области.

Код слишком простой, автор или скрыл основной процесс подготовки features или он настолько элементарный, это нужно проверить.

В статье 3 хорошо документированный код на языке python без зависимостей и абстракций, понятный и структурированный. На изучение кода потребуется два месяца на изучение предметную область и применение.

Хотел бы заметить, что это не адаптация подходов, а применение их с нуля. Потому что, в вашей компании нет накопленных компетенций. С этим может справиться только такой профессионал как я, готовый к вызову любой сложности и способный на полное посвящение делу.

3.3. 3. Предложить план работ.

1. Собрать данные о продукте, которые доступны, в перспективе будут доступны, возможности по экспериментированию, сложность и объем доступных данных.
2. Собрать классические датасеты.
3. Взять самый богатый и популярный из фреймворков, изучить его.
4. Провести первичное сравнение инструментов.
5. Сделать минимальную версию. Протестировать совместно с командой.
6. Изучит методы optimal experimental design (OED)
7. Найти больше датасетов.
8. Выбрать идеальный фреймворк и набор инструментов.
9. Сделать хорошую модель с использованием готовых решений и моделей. Протестировать с командой.
10. Сделать свое решение по последним инновациям, вероятно с

применением языковых моделей и накопленной базой знаний и экспериментов.

Дизайн и экспериментирование, в частности литиевых батарей, это зрелая индустрия, которая породила большое количество open-source инструментов. С которыми, я считаю, необходимо познакомиться, прежде чем пытаться применить последние научные исследований в этой области.

Проекты отсортированы по количеству копий, что показывает их популярность.

- 463f <https://github.com/pybamm-team/PyBaMM>
- 97f <https://github.com/ECSHackWeek/impedance.py>
- 78f <https://github.com/microsoft/batteryml>
- 54f <https://github.com/TRI-AMDD/beep>
- 42f <https://github.com/lionsimbatoolbox/LIONSIMBA>
- 25f <https://github.com/jepegit/cellpy>

3.4. 4. Какие потребуются вычислительные ресурсы или софт.

Через пол года, для 10) пункта в [3.3](#) будет нужен стационарный компьютер за 200-300 тысяч для того, чтобы делать значительные вещи, без которых можно обойтись. Софт и услуги никакие нужны, только железо.

4. Answer eng

As you can see, Article 1 places considerable emphasis on data-based methods, deep learning methods. That's why you need a machine learning software engineer.

4.1. 1. What methods are familiar to me? Which ones can be applied quickly?

Deep learning of neural networks and classical stochastic methods are familiar to me.

Adaptive filters were completely unknown to me, but this is just an application of regression analysis methods, I will study them if necessary.

Stochastic methods can be applied quickly. It would be better to take one of the frameworks and simply use proven models, possibly pre-trained, on modern neural network architectures, without forgetting to evaluate our interpretation capabilities.

4.2. 2. To assess the clarity of the methods and code in articles 2 and 3, to assess the time of adaptation of the approaches.

In article 2, the code is very simple, but it is not documented or organized. It will take about a month to study the code. The difficulty is created by using the proprietary MATLAB language, which I have not used for many years, and my lack of expertise in the subject area.

The code is too simple, the author has hidden the main preparation process features or it's so basic, it needs to be checked.

Article 3 contains well-documented python code without dependencies and abstractions, clear and structured. It will take two months to study the code to study the subject area and application.

I would like to note that this is not an adaptation of approaches, but their application from scratch. Because your company does not have accumulated competencies. Only a professional like me, ready for a challenge of any complexity and capable of full dedication, can handle this.

4.3. 3. Propose a work plan.

1. Collect data about the product that is available, in the future it will be available, experimentation opportunities, complexity and volume of available data.

1. Assemble classic datasets.
2. Take the richest and most popular of the frameworks, study it.
3. To make an initial comparison of the instruments.
4. Make a minimal version. Test it together with the team.
5. Study optimal experimental design (OED) methods
6. Find more datasets.
7. Choose the perfect framework and set of tools.
8. Make a good model using ready-made solutions and models. Test it with the team.
9. Make your decision on the latest innovations, probably

using language models and an accumulated knowledge base and experiments.

The design and experimentation, particularly of lithium batteries, is mature an industry that has spawned a large number of open-source tools. Which, I think, it is necessary to get acquainted with before trying to apply the latest scientific research in this area.

- 463f <https://github.com/pybamm-team/PyBaMM>
- 97f <https://github.com/ECSHackWeek/impedance.py>
- 78f <https://github.com/microsoft/batteryml>
- 54f <https://github.com/TRI-AMDD/beep>
- 42f <https://github.com/lionsimbatoolbox/LIONSIMBA>
- 25f <https://github.com/jepegit/cellpy>

4.4. 4. What computing resources or software will be required.

After half a year, for 10) point B [4.3](#) you will need a 200-300 thousand desktop computer in order to do significant things that you can do without. No software or services are needed, only hardware.

Created: 2024-04-25 Thu 05:31

[Validate](#)