

博士論文

深層学習を活用したリチウムイオン蓄電池の
状態推定技術に関する研究

(Research on lithium-ion battery state estimation
technology using deep learning)

2022年3月

立命館大学大学院理工学研究科
電子システム専攻博士課程後期課程

塩原 孝弘

立命館大学審査博士論文

深層学習を活用したリチウムイオン蓄電池の
状態推定技術に関する研究

(Research on lithium-ion battery state estimation
technology using deep learning)

2022年3月

March 2022

立命館大学大学院理工学研究科

電子システム専攻博士課程後期課程

Doctoral Program in Advanced Electrical, Electronic and
Computer Systems

Graduate School of Science and Engineering

Ritsumeikan University

塩原 孝弘

SHIOHARA Takahiro

研究指導教員：福井 正博 教授

Supervisor : Professor FUKUI Masahiro

目次

論文要旨	i
関連発表論文等.....	iii
第1章 序論.....	1
第2章 準備.....	6
2.1 リチウムイオン蓄電池の構造，動作原理と等価回路モデル.....	6
2.1.1 構造と動作原理.....	6
2.1.2 等価回路モデル.....	9
2.2 従来の状態推定法.....	10
2.2.1 電流積算法.....	10
2.2.2 カルマンフィルタを用いた方法.....	10
2.2.3 深層学習を用いた方法.....	13
2.3 深層学習.....	14
2.3.1 AI と機械学習.....	14
2.3.2 ニューラルネットワーク.....	15
2.3.3 ディープラーニング.....	20
2.3.4 CNN による画像認識.....	23
2.3.5 RNN による時系列データ処理.....	25
2.3.6 転移学習.....	28
2.4 まとめ.....	29
第3章 深層学習を用いた電池構造抽出法.....	31
3.1 はじめに.....	31
3.2 深層学習を活用した電池構造抽出.....	31
3.2.1 ネットワークモデルの構築.....	31
3.2.2 教師データの準備.....	34
3.2.3 提案手法.....	34

3.3	評価指標	35
3.3.1	真陽性率(TPR)と偽陽性率(FPR)	35
3.3.2	ROC 曲線と AUC	36
3.3.3	Intersection over Union (IoU)	38
3.3.4	Pixel Accuracy	38
3.4	本手法の実験方法	38
3.4.1	開発環境	38
3.4.2	データ内容	39
3.4.3	データ前処理	40
3.5	本手法の検証結果と考察	41
3.5.1	実験結果	42
3.5.2	ネットワークモデルの精度比較	45
3.5.3	人による画像抽出	46
3.6	まとめ	48
第 4 章	深層学習を用いた電池状態推定法	50
4.1	はじめに	50
4.2	深層学習を活用した高精度電池 SOC 推定	50
4.2.1	ネットワークモデルの構築	50
4.2.2	教師データの準備	51
4.2.3	知識モデルの活用	52
4.2.4	提案手法	53
4.3	評価指標	55
4.4	本手法の実験方法	55
4.4.1	開発環境	55
4.4.2	データ内容	56
4.4.3	予備実験	58
4.5	本手法の検証結果と考察	58

4.5.1 予備実験結果.....	59
4.5.2 実験結果.....	60
4.6 まとめ.....	67
第5章 結論.....	68
5.1 本研究の成果.....	68
5.2 今後の発展性.....	69
参考文献.....	71
謝辞.....	76

論文要旨

近年、温暖化対策において再生可能エネルギーの利用などによる電力変動の調整機能として蓄電池が期待される。蓄電池をより普及させるためには安全性の向上や小型化・軽量化などの技術力強化が課題であり、対策としてリチウムイオン蓄電池(LIB)が有効である。アプローチには電池の材料開発の効率化と内部状態の推定技術の高精度化があげられる。情報科学を活用して材料開発を強化するマテリアルズインフォマティクス(MI)の研究が進められている。シミュレーション技術の進化により開発効率化が進む中で、物体の位置形状情報の入力データの準備に多くの時間を費やしており、この自動化が課題として残る。また電池の内部状態の推定技術の高精度化において、電池残量(SOC)推定は最も重要かつ基本的な技術であり高精度化が求められる。

本論文は、深層学習を活用して、LIBの走査型電子顕微鏡(SEM)画像データから空隙箇所を自動抽出する作業の効率化と、蓄電池内部の状態管理に必要なSOC推定の高精度化を目的とする。深層学習の活用には、適切なネットワークモデルの構築、教師データの準備と、学習時間の短縮が大きな課題として認識される。

まず畳み込みニューラルネットワーク(CNN)の中のU-Netモデルのネットワーク構造を進化させ、学習効率に悪影響を及ぼす内部共変量シフトを抑えるため特徴量マップにバッチノーマライゼーションを加え、畳み込み層数やフィルタ数、画像のならし効果やロバスト性向上のためプーリング条件などを最適化した。さらに構成を左右対称にしてアップサンプリングで特徴マップをクロップせずそのまま利用した。これらの特徴を持つ独自のネットワークモデルを構築した。同モデルを使用してSEM画像データから空隙情報を自動抽出する方法を提案する。熟練者が作成した空隙抽出データと予測結果のピクセル単位の重なりからIoU(Intersection over Union)評価指標で精度比較した。被験者は初心者(入社2年)から熟練者(入社25年)までの材料開発者10名で構成する。経験の違いにより被験者間で精度に2~5倍の差が確認できたことから、空隙を見分けるには電池材料の立体的構造がイメージできる経験に裏付けされた判断が必要であると考えられる。被験者のIoU精度の平均値(mean)30.1%に対して本手法は約1.5倍の44.4%で抽出した。熟練者の最も高い精度49.2%で熟練者のレベルを超えることはできなかったが、かなり近い精度を実験的に確認できた。抽出作業は1画像当たり人手で1時間程度、本手法は数秒程度で完了した(学習時間除く)。本手法のIoU精度は、標準FCNモデル30.3%、FCN改モデル52.7%に比べ67.5%であり、明らかに精度に優位性が見られた。以上から本手法による空隙抽出の自動化による作業効率向上への有効性が確認され工学的に有用であると考えられる。

次にリカレントニューラルネットワーク(RNN)の1つであるLSTM(Long Short-term Memory)モデルに対してデータ次元数や同時処理データ分割長、隠れ層数の設定、学習率などのパラメータ条件や処理手順を最適化して独自のネットワークモデルを構築した。同モデルを使用してBMSで観測される端子電圧、電流値から電池のSOC推定を行う方法を

提案する。一方、教師データの準備方法について、まず実験室で諸条件を設定した等価回路モデルによって得られた測定値を教師データに用いた。次に実際の観測データから統計的推定手法(RLS-EKF)を用いて得られた SOC 推定値を教師データに充足することで多様な教師データを準備可能にした。これらの特徴を持つ革新的な独自の教師データ構築方法を提案した。SOC の実測値と推定値の差の平均平方二乗誤差(RMSE)評価指標で精度比較した。EKF 法では放電パターンの違いにより約 0.5~3.0%で精度にバラツキが生じるが、本手法はすべてのデータで約 1.0%前後の安定した精度を実験的に確認した。さらに過去に学習した知識モデルを活用した転移学習(ファインチューニング)により教師データ数が半分でも精度 1.0%前後を維持し、学習時間では約 2200 秒から約 1000 秒に短縮され 55%減の転移学習の効果を確認できた。

実験により得られた成果は、本提案手法におけるネットワークモデルと教師データ準備方法と学習効果によるものであり、革新性と有効性の高さの可能性を示した。

関連発表論文等

論文誌等採録論文

- [1] 塩原孝弘, 内野学, 福井正博, “ディープラーニングを用いた画像解析によるリチウムイオン電池における空隙情報の自動抽出方法の探索,” 電子情報通信学会 和文論文誌 C, Vol.J103-C, no.3, pp.143-152, 2020年3月.
- [2] 塩原孝弘, 福井正博, “統計的推定手法を併用した教師データの構築と転移学習に基づく LSTM による残量推定の高効率高信頼化,” 電子情報通信学会 和文論文誌 B, Vol.J105-B, no.3, 2022年3月. (採択済, 公開予定)

国内学会・シンポジウム等における発表

- [1] 常田健斗, 塩原孝弘, 福井正博, “適応型カルマンフィルタにおける初期値設定戦略に関する一考察,” 電子情報通信学会 回路とシステム研究会 (CAS), オンライン開催, 2021年1月.
- [2] 川瀬慎太郎, 塩原孝弘, 福井正博, “蓄電池充電状態推定のためのリカレントニューラルネットワーク適用実験と評価,” 電子情報通信学会 回路とシステム研究会 (CAS), オンライン開催, 2021年1月.
- [3] 鈴木慶一郎, 太田秀星, 塩原孝弘, 福井正博, “カルマンフィルタを用いたリチウムイオン蓄電池充電率推定精度向上の検討,” 電子情報通信学会 回路とシステム研究会 (CAS), オンライン開催, 2021年10月.

第1章 序論

温暖化対策が経済成長の制約やコストではなく成長の機会と捉える時代となり、日本では2020年10月に2050年カーボンニュートラルが宣言された。経済と環境の好循環に繋がるよう従来の発想を転換し、イノベーションを起こす前向きな挑戦が求められている[1]。

特に電力に関する脱炭素化が大きな課題であり、自動車の電動化や再生可能エネルギーを最大限導入するのに必要となる電力変動の調整力として、蓄電池が期待される。自動車の場合、従来のガソリンなど化石燃料を利用する内燃機関の自動車ではエネルギー効率 20～30%で運動エネルギーに変換されるが、電気を利用する電気自動車(Electric Vehicle, 以下 EV と表記)では 90%以上の効率で変換され、CO₂などの排出を低減することができる。エネルギー密度の高い EV 用蓄電池や定置用蓄電池をより普及させるためには、安全性の向上や小型化・軽量化などの技術力強化が課題である。

非常に軽くてイオン化傾向が大きい元素であるリチウムを用いることで、リチウムイオン電池(Lithium-ion Battery, 以下 LIB と表記)は、従来から使われている鉛蓄電池、ニッケルカドミウム電池やニッケル水素電池などの蓄電池と比べて、高い電圧とエネルギー密度を持つ[2]。LIB は小型で軽量の電池をつくる上でメリットが大きく、EV や電力網の平準化のための定置用電源などに多く用いられており重要性は高い。本論文では、これらの電池の中で主流となっている LIB を対象とする。

この LIB のさらなるエネルギー密度向上と安全性の向上の両立を目指して、世界各国各社で電池材料開発競争にしのぎを削っている。材料開発において、新しい材料および試料配合などの組み合わせや手順などの設計条件を決めて実験し、結果を評価して新材料や代替材料を探索する。従来の材料開発の方法は、研究者個人の経験と勘によって新たな材料を合成し、その材料の特性を調べることを繰り返し進めてきたが、網羅的な実験の繰り返し回数には限界があり、開発速度を上げることが困難である。また革新的な材料は偶然発見したケースであることが多く、属人的かつ偶発的な要素が多分に含まれる。材料開発において一つの新材料を見つけるのに 10 年以上を要することも稀ではない。

近年、開発効率化を進めるため、米国では国家プロジェクト Materials Genome Initiative[3]が 2011 年から始まり、その後世界中でマテリアルズインフォマティクス(Materials Informatics, 以下 MI と表記)の研究が進められ、欧州では Novel Materials Discovery[4]が 2015 年から始まり、開発競争が激化している。MI とは、大量のデータからデータ間の相関関係や隠れた規則性などを見出すデータマイニングなどの情報科学を通じて、求める特性値(機能, 性能)を持つ新材料や代替材料を効率的にかつ非属人的に探索する取り組みである。既にマサチューセッツ工科大学とサムスン電子社が共同で、実験せずにシミュレーションと計算機科学を用いた MI により、LIB の固体電解質の新材料を発見し注目された[5][6]。日本でも 2014 年内閣府の「戦略的イノベーション創造プログラム(SIP)」[7]でマテリアルズインテグレーションと呼ばれる同様の国家的な取り組みが始まった。

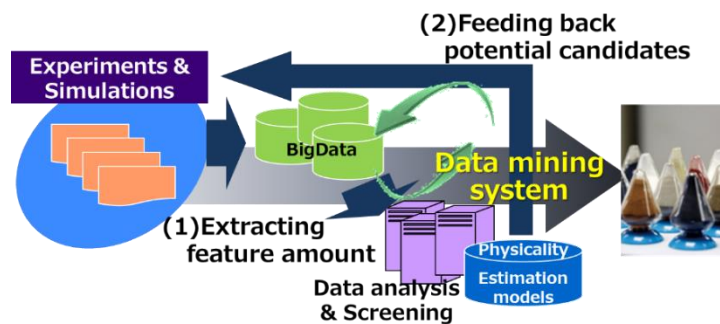


図 1-1 マテリアルズインフォマティクス概略図

Fig.1-1 A structure of materials informatics.

図 1-1 に MI を進める手順の概略図を示す. MI は大きく 2 つのアプローチに分けられる. まず, (1) LIB などの走査型電子顕微鏡(Scanning Electron Microscope, 以下 SEM と表記)などの画像データから注目する実験試料などの情報を抽出し数値化してデータベースに格納する. 次に(2)同データと特性値との関係性を見出し, 目標特性値からの逆問題として, 最適な材料の組み合わせや実験条件を導き出す. フィードバックされた材料の組み合わせや実験条件を用いて材料を合成し, その材料の特性を調べるといったサイクルを繰り返す. 人の知識・経験・能力・感性に依存し, あるいは網羅的な条件での膨大な実験回数による方法から, 人に依存せずにシミュレーション技術を活用し, データ分析により導出される確度の高い条件での実験方法を選択するといった材料開発効率を向上させるシステムである. この時, 注目する実験試料の混合・分散・配合の具合や大きさと分布の構成比, 立体的構造をシミュレーションにより確認する前処理として, 対象とする材料を入力画像から領域分割してデジタルデータに変換する(1)の作業に多くの時間を費やしている. しかも担当者の熟練度で抽出精度に大きな差が生じてしまうため, 安定して高い抽出精度と前処理作業の効率化が求められる.

画像解析に適用するデータは, 電池試料をスライスした断面を SEM によって撮影された LIB の内部構造の画像である. 図 1-2 に LIB 正極断面の模式図を示す.

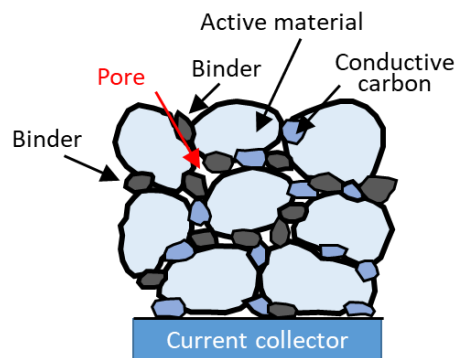


図 1-2 LIB 正極の模式図

Fig.1-2 Schematic image of electrode reaction in LIB Cathode.

LIB は活物質、電解液、バインダー、導電材、セパレータ、電極から構成され、電極は正極と負極の二つで構成される。LIB 電極は電極反応を起こす活物質、電極の電子伝導性を向上させる役割を持つ導電助剤、及び各材料を結着し電極形状を保持するためのバインダー樹脂を Al や Cu の集電体上に塗工して形成される。電池内空隙(pore)部に電解液が充填される。

シミュレータでは多くの SEM 画像を連続的に重ね合わせて、隣接する情報を 3D モデル立体構造に組み上げて三次元観察を行うことができる。図 1-3 に立体構造の模式図を示す。シミュレータで原子配列のような物性の特性を状態認識し、コンピュータ上で計算して特性を解析して材料探索を進める。同じ材料や配合であっても各種材料の混合状態・接触状態を制御することが重要になる。空間認識した情報を扱うことで材料構成やその製造方法を予測し実現可能性を探ることができる[8]。

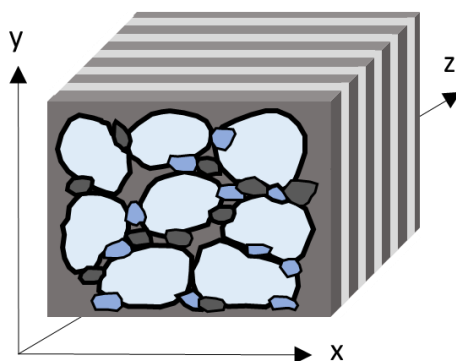


図 1-3 LIB 正極の立体的構造の模式図

Fig.1-3 Schematic 3D image of electrode reaction.

シミュレーション技術の進化により開発効率化が進む中で、事前準備としてシミュレータへの材料の位置形状情報の入力データ作成に多くの時間を費やしており、この自動化が課題である。これらのシミュレータ環境を活用し有効に機能させるには、画像認識技術を活用して LIB の構成物質の内部構造の立体的な状態を画像データから正確に位置形状情報を抽出して数値化し、正しい情報をシミュレータに伝える必要がある。最近では SEM 画像撮影から直接 3D シミュレータに入力できる装置が市販され始めたが、導入における装置環境やツールの精度の合わせ込みの調整において、精度の信頼性検証に手間がかかるため、現状では材料開発を効率的かつ有効に進めるまでの実用には課題が残る。

本論文では、MI における(1)前工程について、SEM 画像を解析して LIB の構成物質を自動抽出し数値化することを目的とし、空隙情報を抽出対象とした。空隙は解析時に電池内で空洞になっている部分であり、この部分に電解液が多く溜まっていた領域を表す。空隙は電池の内部状態や電池特性を解明するのに重要な情報になる。

次に電池の内部状態の推定技術の高精度化について検討する。電池は時間の経過や充放電頻度などの使用状況により性能劣化が進む。電池の劣化とは電池を使用することで徐々にその性能を低下させていく現象を指す。電池の劣化は内部抵抗が増加し電池容量が減少するため、機器の使用時間や EV の航続距離の減少につながる。また電池には、放電し過ぎた場合の過放電、充電し過ぎた場合の過充電というモードがあり、この状態を繰り返すと電池の劣化が急速に進む[9]。電池状態を管理するシステムを電池管理システム(Battery Management System, 以下 BMS と表記)と呼ぶ。BMS では各種センサを用いて電池の電流と電圧、温度を監視し、これらの観測値から電池の残量(State of Charge, 以下 SOC と表記)や健全度(State of Health, 以下 SOH と表記)の推定値を用いて、充放電制御やセルバランス制御、電池の劣化診断、漏電等の異常検知など電池状態の維持・監視を行う。LIB に蓄えられた電気エネルギーを効率的かつ安全に利用するには、安全設計に基づいた BMS が必要不可欠である。BMS を正しく機能させるためには、BMS の電池状態推定の中で重要事項の 1 つである SOC のレベルの高い推定精度が求められる。

最近では、機械学習を使用したデータ駆動型アプローチが実現可能になり、これまでの統計的手法に比べて煩雑なモデリングの労力が削減され、同等またはそれ以上の精度でソリューションを導き出す可能性に期待されている。機械学習において深層学習が注目され、様々な分野で画像認識技術や推定技術に活用されてきており、電池の推定方法として、その有用性に期待が高まっている。

以上のことから、本論文では、深層学習を活用して LIB の SEM 画像データから空隙箇所を自動抽出する作業の効率化と、蓄電池内部の状態管理に必要な SOC 推定の高精度化を目的とし、まとめたものである。

本論文の構成を以下に示す。

第 1 章では、本研究の背景として、電力変動の調整機能として蓄電池が注目され、その中でも高いエネルギー密度を持つ LIB が有効であると期待され、さらなるエネルギー密度向上と安全性の向上を目指して電池材料開発の競争が激化している。材料開発の効率化にマテリアルズインフォマティクス(MI)の研究が進められ、シミュレータで電池内部構造の立体的な状態をとらえるため、画像データから物体情報抽出作業の高精度・高効率化が求められる。また LIB に蓄えられた電気エネルギーを効率的かつ安全に利用するには、安全設計に基づいた BMS を正しく機能させる高精度な SOC 推定技術が必要であることを説明した。

第 2 章では、本提案手法を説明する準備として、対象とする LIB の構造と動作原理、回路モデルを説明した。次に先行研究事例を調査し、本提案手法で利用される統計的手法や深層学習の基本理論について説明した。さらに深層学習を活用するには、最適なパラメータ条件設定を含む「適切なネットワークモデルの構築」や、課題に対処するために適切な質、量の「教師データの準備」と、運用方法も含めた「学習時間の短縮」が大きな課題として認識されることを説明した。

第 3 章では、深層学習を活用して、畳み込みニューラルネットワーク(CNN)の中の U-Net モデルから新たな独自ネットワークモデル構造を構築し、このモデルを用いて SEM 画像データから空隙情報を自動抽出する方法を提案した。本手法では U-Net モデルネットワーク構造をもとに、学習効率に悪影響を及ぼす内部共変量シフト対策に加えてフィルタ設計などでロバスト性向上を図り、特徴マップを活用したアップサンプリング処理を加えた独自のネットワーク構造を提案した。初心者と熟練者といった被験者の経験レベルによって抽出精度にかなりの差が生じることが確認できた。本手法は熟練者の物体抽出精度に近いレベルにまで精度を高めることができた。人による抽出に対する本手法の精度の到達位置を実験結果から確認することができ、本手法の優位性と工学的に有用であることを確認することができた。

第 4 章では、深層学習を活用して、リカレントニューラルネットワーク(RNN)の中の LSTM モデルを用いて観測データ(端子電圧, 電流)から電池の SOC 推定を行う方法を提案した。深層学習で有効な結果を得るためには教師データをいかに準備するかが課題である。本手法では、まず実験室で諸条件を設定した等価回路モデルによって得られた測定値を教師データに用いた。次に実際の運用時に観測データから統計的推定手法(RLS-EKF)を用いて得られた SOC 推定値を教師データに充足することで、多様な教師データの準備が可能になる独自の教師データ構築方法を提案した。SOC 推定精度について教師データ数の違いによる影響、EKF 法との精度比較、過去に学習した知識モデルを使った転移学習による精度と計算時間の比較による学習効率への効果を確認した。それぞれの実験で良好な結果が得られ、革新的な教師データの準備方法を含む、本提案手法による SOC 推定の有効性を示した。

第 5 章では、本研究成果のまとめを紹介した。さらに本提案手法を通じて、今後は教師データが少なくても学習効率を向上させて電池材料開発を加速させる取り組みや、さらなる高い精度を示す SOC 推定への検討課題を説明した。将来的に電池状態の異常検知や不具合検知へと発展させ、将来の電池の安全かつ有効な利用へとつなげるよう今後の展開への可能性を提示した。

第2章 準備

本章では、提案手法の準備として、まず基本的な LIB の構造と動作原理について述べる。次に本論文で扱う電池の状態について定義し、回路モデルを設定する。次に従来の電池の SOC 推定方法に使用される代表的な電流積算法とカルマンフィルタ法を説明する。さらに最近注目されている機械学習におけるディープラーニングについて説明する。様々な課題に対応する代表的な、畳み込みニューラルネットワーク (Convolutional Neural Networks, 以下 CNN と表記) と、リカレントニューラルネットワーク (Recurrent Neural Networks, 以下 RNN と表記) について、データの特徴に合わせた画像認識や自然言語処理に適用する方法を説明する。さらに学習効率を高める 1 つの方法として、過去の知識モデルを活用する転移学習について説明する。

2.1 リチウムイオン蓄電池の構造、動作原理と等価回路モデル

本節では LIB の構造や動作原理と状態について説明し、これを表現する等価回路モデルを定義する。

2.1.1 構造と動作原理

LIB は正極、負極、集電体、電解液、セパレータなどで構成される[10]。図 2-1 に示すように、負極(negative electrode)と正極(positive electrode)が電解質(electrolyte)を介して対向している。電池内の空隙(pore)部に電解液が充填されている。両電極は一般に粒子形状の活物質(active material)、主に炭素微粒子が利用される導電助剤(conductive carbon)、粒子間の構造を保つためのバインダー(binder)と、電極合剤層が集電体(current collector)上に形成されている。

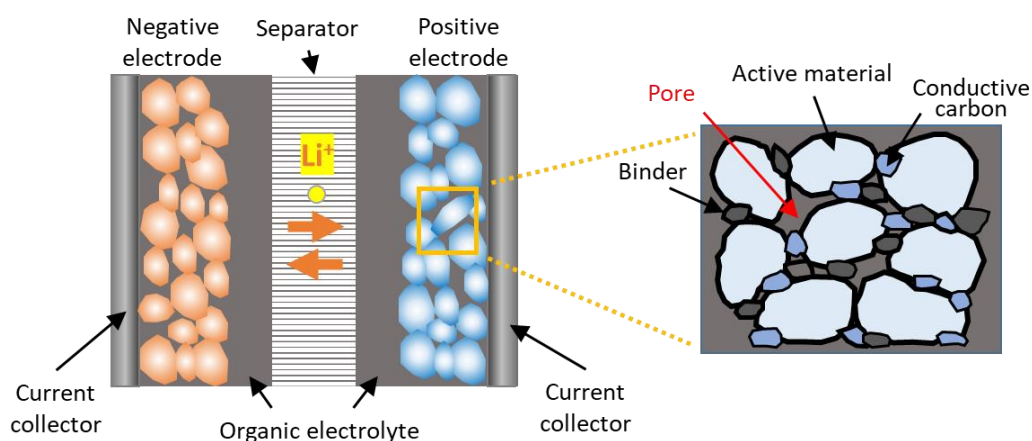


図 2-1 リチウムイオン蓄電池の構成と動作原理[12]

Fig.2-1 Configuration and operating principle of Lithium-ion Battery.

蓄電池は電気エネルギーを化学エネルギーに貯蔵し、電気エネルギーに変換するデバイ

スである。リチウムイオンが電解液を介して正極と負極の間を行き来することで充放電が行われる[11]。充電時では電流の移動に伴い正極からリチウムイオンが電解液に抜け出して負極に移動する。逆に放電時では負極からリチウムイオンが電解液中に抜け出して正極に移動することで外部回路に電流が流れる仕組みである。低温での急速充電時は、負極材の表面でリチウムが金属化して析出することがあり、析出した金属が正極と接触すると内部短絡を起こし、発熱や爆発する恐れが生じる。このような正極と負極の内部短絡を防ぐために正極と負極の間にセパレータを入れて絶縁する。

電池試料をスライスして断面の SEM 画像を撮影する際に、電解液を抜くことで空洞になる。空隙は電池内の空洞になっている部分を指し、電解液が溜まっていた領域を表す。一般的に粒子の比表面積が大きいほど電解液との接触面積が大きくなり、反応が進みやすいため、粒子の大きさとその分布が重要になる。さらにこれら粒子形状など内部構造もイオンの通り道（導電パス）に関連するため電気伝導性に相関を持つ[12]-[14]。電極内の活物質などの粒子間の空隙はリチウムイオンの通り道となる（図 2-2）。そのため空隙構造は電池の性能を左右する重要な情報となる。

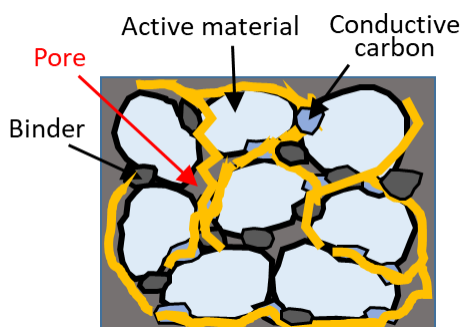


図 2-2 イオンの通り道

Fig.2-2 Pores network structure.

同じ材料、配合でも粒径調整や立体空間構造を意図的に組み上げていく必要がある[15]。より高性能な電池を開発するためには、空隙構造を把握し、材料や添加剤などの混合・分散・複合や粒径などを考慮して、電極内でのイオンの通り道を緻密に制御することが求められる[16]。すなわち材料特性、大きさと分布、粒子形状や空隙構造など立体空間構造の把握が、導電パスといった電気伝導性と関連して電池性能の向上につながる。LIBの電極において、活物質粒子単体では内部抵抗が大きいため、活物質中の電子を効率よく伝えるために活物質の周りに導電助剤を添加して活物質粒子間に介在させて導電性を上げて導電パスを確保する。例えば正極でコバルト酸リチウム粒子にカーボンナノ粒子のような導電材の微粉体を添加し、活物質の周りにカーボンナノ粒子の薄い層を形成できるかどうかで電池特性が大きく変わる。粒子や活物質の大きさや、活物質などの粒子に付着する導電助剤やバインダーの付着状況、混合状態など、粒子の位置関係や空隙領域などの内部構造を把握し、意図的に電子やリチウムイオンの通り道を生成することが重要である。リチウムイオンの通り易

さ、製造手段における内部構成の実現性や、導電ネットワークを生成し制御する内部構成の設計には、立体空間構造を把握することが極めて重要である。

電池が劣化すると内部抵抗の増加により出力が低下し、電池容量が減少する(図 2-3)。

まず内部抵抗には、正極表面の結晶構造の変化による抵抗、正極・負極へリチウムイオン挿入・脱離する際の電荷移動抵抗、活物質内での拡散抵抗、電解液中の電荷移動抵抗などがある。内部抵抗が増加する要因の一つとして、LIBは充放電動作時に正極、負極共に電極構造材の隙間にリチウムイオンが入り出すインターカレーション反応があり、この時に活物質自体の膨張収縮が起こる。電極はバインダーで集電体などに接着されている。電池の充放電サイクルが繰り返されると膨張収縮からの物理的圧力により、これに接着するバインダーが破壊されて結着性が次第に弱まり切れてしまう。活物質間や活物質-導電助剤間、活物質-集電体間などのあらゆる結合が弱まり、導電パスが切断され抵抗が大きくなる。内部抵抗が増加することで出力低下が起こる。

次に電池容量を減少させる要因には、負極表面のリチウムイオンが電解液成分とともに負極活物質表面に皮膜として定着するSEI(Solid Electrode Interface, 以下SEIと表記)の生成による容量劣化がある。さらにリチウムイオン金属析出は容量劣化要因になる。キャリアとなる正極内のリチウムイオンが損失してしまい電極間を移動するリチウムイオンの量が低下する。一方ではSEIは負極と電解液の界面において電解液の分解を防ぎ、リチウムイオン挿入・脱離を容易にする役割も持つ。負極表面の電位低下はSEI生成によって起きやすくなりSEIの進行がリチウムイオン金属析出の発生を早めることも考えられることから、SEI形成を制御する必要がある、さらに異常に対する予兆管理が求められる。

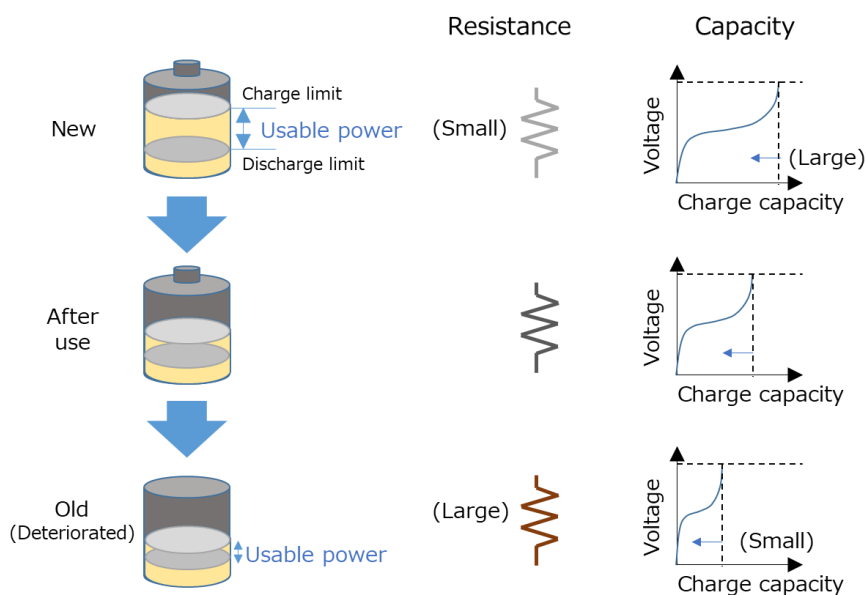


図 2-3 電池の劣化

Fig.2-3 An image of battery degradation.

蓄電池の充放電反応の基本原理の数理モデル化には、電気化学モデルや等価回路モデルが提案されている。電気化学モデルは正確で精度がいいが、設定すべき初期条件、境界条件や、パラメータが多くあり設定が難しい。しかも主反応以外に副反応を考慮する場合はさらに式が複雑になり計算量が多くなる。一方の等価回路モデルは、電気化学の専門知識を必要とせず、モデルが単純で扱いやすい。電気化学モデルに比べて計算量が少ない。しかし電池を正確に表現できる範囲が狭く、しかもパラメータ特性の調整に時間が掛かるという欠点がある。そのため等価回路モデルによる推定では、実験的な温度特性を温度条件別にテーブル形式あるいは数式の設定といったパラメータ調整により推定精度を高める工夫が必要である。本論文では、外部から測定可能な端子電圧と電流との関係が明確で扱いやすい等価回路モデルを採用し、LIBのSOCを高精度に推定する方法を提案する。

2.1.2 等価回路モデル

等価回路モデルは蓄電池の電気的特性を等価な電気回路素子で示したモデルである。特にRC等価回路モデルはモデルベースのSOC推定に広く利用されている。Plettのフォスター型等価回路[17][18]を用いて、LIBの等価回路モデルを図2-4で表現する。フォスター型等価回路の特徴はLIB内の3つの物理化学現象、①電解液中の電荷移動抵抗、②正極・負極へリチウムイオン挿入・脱離する際の電荷移動抵抗、③活物質内での拡散抵抗、について直列接続の形式で表現できる点にある。

LIBの内部抵抗は電氣的で高速な反応（電荷移動抵抗）と電気化学的で遅い反応（拡散抵抗）の複合である。前者は数ミリ秒、後者は1秒～数時間の遅い時定数反応を示す。本論文では、等価回路モデルは電解液中の電荷移動抵抗を表す直流抵抗 R_a 、活物質内での拡散に伴う遅い反応を表す一段抵抗とコンデンサの並列回路(R_b, C_b)を内部抵抗とし、無負荷状態の蓄電池電圧を意味する開放電圧 U_{ocv} のOCV(Open Circuit Voltage, 以下OCVと表記)で構成される。OCVは理想電源で模擬され、 U_b はRC回路(R_b, C_b)の両端電圧、 U_L は蓄電池の端子電圧、 I は電流である。電池に流れ込む方向を正方向に定義し、 I の向きは放電には負、充電には正の値として計算する。

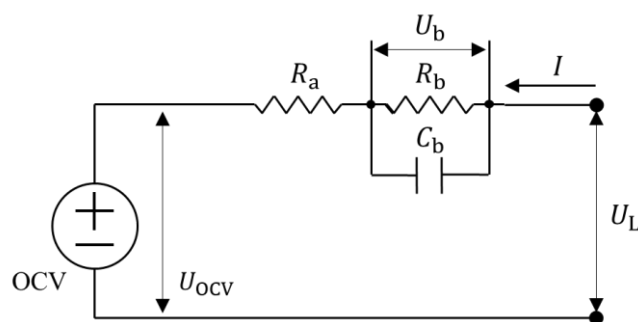


図 2-4 LIB 等価回路モデル

Fig.2-4 LIB equipment circuit model.

2.2 従来の状態推定法

本節では、SOC 推定手法の従来技術として代表的な、電流積算法[19]、カルマンフィルタを用いた方法[20]-[22]、深層学習を用いた方法[23]-[25]について紹介する。蓄電池の SOC は利用可能な容量がどれくらい残っているかを示す重要な電池指標であり、SOC 推定は最も重要かつ基本的な技術である。

2.2.1 電流積算法

電流積算法[19]は広く用いられる SOC 推定手法である。通過する電流を積算することで電池に入出力する電荷量をもとめる。時刻 t 秒における電荷 q と電流 I の間には、式(2.1)で表される関係があることを利用する。

$$q(t) = \int_{t_0}^t I(\tau) d\tau \quad (2.1)$$

SOC は現在の電池の電荷量を電池の満充電容量で除算して求められる。推定開始時刻 t_0 秒から時刻 t 秒における SOC を式(2.2)で表す。

$$SOC(t) = SOC(t_0) + \frac{1}{FCC} \int_{t_0}^t I(\tau) d\tau \quad (2.2)$$

$SOC(t_0)$ は SOC の初期値、 FCC は満充電容量(Full Charge Capacity)を表す。 $I(t)$ は時刻 t における充放電時の電流である。電流積算法は、高精度な電流センサを利用すれば、高精度な SOC 推定を行うことができるが、SOC の初期値誤差があれば誤差がオフセットとして残る。さらに電流センサに誤差があれば誤差が累積するため推定精度は劣化する。また内部で消費される自己放電は計測できない。電池劣化で FCC が減少することによる誤差が生じるなどの欠点を持つ。これらの問題は情報をフィードバックして補正していないために生じる。

2.2.2 カルマンフィルタを用いた方法

カルマンフィルタは、状態空間モデル(State Space Model)において数理モデルで表現され、内部の見えない状態を効率的に推定するモデルベースの計算手法である。同手法を蓄電池に適用する場合、観測可能な端子電圧と電流を用いた SOC 推定は、状態空間モデルの内部状態の推定問題として扱うことが可能である。

高次の非線形モデルを用いたカルマンフィルタ法では、拡張カルマンフィルタ(Extended Kalman Filter, 以下 EKF と表記)法[20]が提案されている。電池の端子電圧、電流、内部抵抗、分極による端子電圧の過度現象をモデル化して EKF で処理する SOC 推定方法は、高精度かつリアルタイム性の良い予測方法として広く認知されている。しかし電池の使用条件によってモデルで定義する回路パラメータが時々刻々変化するため、推定精度に影響を

与える。またシステムノイズと観測ノイズの設定値次第で推定精度に大きく影響する。その他 Plett ら [18], [20], [26] は蓄電池の内部パラメータと EKF による SOC の同時推定を行い SOC 推定の高精度化を実現している。また Sigma Point カルマンフィルタによる SOC 推定の高精度化 [27][28] も進められているが、EKF を適用する場合に必要なノイズの設定が課題として残る。図 2-4 で示した LIB 等価回路モデルを微分方程式で表現すると式(2.3)になる。

$$C_b \frac{dU_b}{dt} + \frac{U_b}{R_b} = I \quad (2.3)$$

$$U_L = U_b + IR_a + U_{ocv}$$

LIB の SOC 推定に離散時間状態空間モデルを選択する。サンプリング時間を Δt 、離散値 k として式(2.3)に前進オイラー法で離散化した差分方程式で表現すると式(2.4)になる。

$$U_b(k+1) = \left(1 - \frac{\Delta t}{R_b C_b}\right) U_b(k) + \frac{\Delta t}{C_b} I(k) \quad (2.4)$$

$$U_L(k) = U_b(k) + I(k)R_a + U_{ocv}(k)$$

式(2.4)を整理すると以下の式(2.5)になる。

$$U_L(k) = R_a I(k) + \left(\frac{\Delta t R_a}{R_b C_b} + \frac{\Delta t}{C_b} - R_a\right) I(k-1) + \left(\frac{\Delta t}{R_b C_b} - 1\right) U_{ocv} - \left(\frac{\Delta t}{R_b C_b} - 1\right) U_L(k-1) + U_{ocv} \quad (2.5)$$

ここで $I(k-1)$ 、 $U_L(k-1)$ は入力変数、それ以外はパラメータ（定数）のため、式(2.6)のように変数変換して書き換えることにより式(2.7)を得る。

$$b_0 = R_a, \quad b_1 = \frac{\Delta t R_a}{R_b C_b} + \frac{\Delta t}{C_b} - R_a, \quad (2.6)$$

$$a_1 = \frac{\Delta t}{R_b C_b} - 1, \quad f = (1 + a_1)U_{ocv}$$

式(2.7)は未知のパラメータ $\theta = [b_0 \ b_1 \ a_1 \ f]^T$ とする回帰式である。

$$U_L(k) = \varphi^T(k)\theta$$

$$\varphi(k) = \begin{bmatrix} I(k) \\ I(k-1) \\ -U_L(k-1) \\ 1 \end{bmatrix}, \theta = \begin{bmatrix} b_0 \\ b_1 \\ a_1 \\ f \end{bmatrix} \quad (2.7)$$

回帰式から θ を求めることで、回路パラメータ R_a, R_b, C_b, U_{OCV} は式(2.8)で計算される。

$$R_a = b_0, \quad R_b = \frac{b_1 - a_1 b_0}{1 + a_1},$$

$$C_b = \frac{\Delta t}{b_1 - a_1 b_0}, \quad U_{OCV} = \frac{f}{1 + a_1} \quad (2.8)$$

非線形システムの離散時間状態空間モデルを状態方程式 (2.9)、観測方程式 (2.10)で表す。 $u(k) = I(k)$ がシステムへの入力、 $x(k)$ が状態、 $y(k)$ が観測値を表す。EKF を用いた方法では、システムノイズ w_k と観測ノイズ v_k が存在し、平均 0 の正規ノイズで互いに独立であると仮定する。

$$x(k+1) = Ax(k) + BI(k) + w_k \quad (2.9)$$

$$y(k) = Cx(k) + v_k \quad (2.10)$$

次に状態ベクトルと観測ベクトルを式(2.11)、各回路パラメータを式(2.12)に示す。 $x(k)$ を SOC, U_b の集合として定義し、 $SOC(k)$ は時刻 k の残量、 $OCV(SOC)$ は開放電圧、 FCC は蓄電池の満充電容量である。

$$x(k) = \begin{bmatrix} SOC(k) \\ U_b(k) \end{bmatrix} \quad (2.11)$$

$$y(k) = U_L(k) = OCV(SOC) + I(k)R_a(k) + U_b(k) + v_k$$

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 - \frac{\Delta t}{R_b C_b} \end{bmatrix}, B = \begin{bmatrix} \frac{\Delta t}{FCC} \\ \frac{\Delta t}{C_b} \end{bmatrix}, \quad (2.12)$$

$$C = \begin{bmatrix} \left. \frac{dOCV}{dSOC} \right|_{SOC=\widehat{SOC}} & 1 \end{bmatrix}$$

このように逐次最小二乗法(Recursive Least Square, 以下 RLS と表記)により回路パラメータを推定し、EKF のパラメータとして伝達され、EKF で状態(SOC)を計算することで、内部抵抗や OCV の時間的変化に追従して状態を推定する[21][22]。

EKF は変化する内部状態を高精度に推定し、観測値により補正する。具体的には予測ス

ステップで状態空間モデルを用いて時間 k の状態から時間 $k + 1$ の状態を予測し、フィルタリングステップで平均二乗誤差が最小となるよう予測ステップで得られた時間 $k + 1$ の状態を補正するという処理を繰り返す。予測ステップによって得られた状態の予測値を事前推定と呼び、推定を意味する記号 $\hat{\cdot}$ と事前を意味する記号 $^-$ を使い $\hat{x}(k)^-$ と表記する。また補正ステップ後の状態推定値を事後推定とよび、 $\hat{x}(k)$ と表記する。状態推定誤差の共分散行列を P_k^- 、システムノイズ w_k の分散 $E[w_k w_k^T]$ を σ_w^2 、観測ノイズ v_k の分散 $E[v_k v_k^T]$ を σ_v^2 とし、これらの初期値を最尤推定法などにより与える。以下の Algorithm I 式(2.13)~(2.17)で計算し、状態ベクトル $\hat{x}(k)$ から SOC 推定値を得る。この時、 σ_w^2 、 σ_v^2 は平均0の正規ノイズで互いに独立であると仮定する。 E は 4×4 の単位行列とする。

Algorithm I

<初期値設定>

$\hat{x}(0), P(0), \hat{\sigma}_w^2, \hat{\sigma}_v^2$ の設定

<予測ステップ>

$$\hat{x}^-(k+1) = A\hat{x}(k) + BI(k) \quad (2.13)$$

$$P^-(k+1) = AP(k)A^T + \sigma_w^2 \quad (2.14)$$

<フィルタリングステップ>

$$g(k+1) = \frac{P^-(k+1)C^T}{CP^-(k+1)C^T + \sigma_v^2} \quad (2.15)$$

$$\hat{x}(k+1) = \hat{x}^-(k+1) + g(k+1)\{y(k+1) - \hat{y}^-(k+1)\} \quad (2.16)$$

$$P(k+1) = (E - g(k+1)C)P^-(k+1) \quad (2.17)$$

EKF で処理する SOC 推定方法は、高精度かつリアルタイム性の良い予測方法として広く認知され、しかも蓄電池の内部パラメータと SOC の同時推定を行い SOC 推定の高精度化を実現している。しかしながら一般に SOC 同時推定法は複雑な非線形性を有するため、精度の安定性の向上が難しい。またシステムノイズと観測ノイズのそれぞれの分散値の最尤推定値は、EKF 使用中に最適値が変動する可能性があり、さらなる精度の安定性の向上が必要とされる。このことからノイズ情報の最適値設定を不要とし、かつ EKF 使用中に最適値に追従させるために、観測誤差からシステムノイズや観測ノイズを自動で更新してノイズ適応する適応型拡張カルマンフィルタ AEKF(Adaptive Extended Kalman Filter, 以下 AEKF と表記)手法[29]-[31]が提案されている。しかし同手法を用いて SOC 推定を行うには、ノイズ情報の更新を開始するタイミングやノイズ適応するタイミングを判断する条件などが SOC 推定精度に大きく影響するため、条件設定に多くの研究課題が残っている。

2.2.3 深層学習を用いた方法

深層学習を用いた方法[23]-[25]では、様々な充放電パターンや周囲温度条件下での SOC 推定に対して異なるパラメータ設定やモデルを必要としない。少ない周囲温度条件の学習

条件で電池状態の SOC 推定をしているが、高精度な推定を維持するには、実際の運用条件に合った適切な周囲温度条件の測定データから教師データを一定数準備する負担が残る。また転移学習[32]により学習効率化による精度向上と計算時間削減効果が確認されているが知識モデルの転移元との相性など調整事項が多い。深層学習の特徴やアルゴリズムについては、次節で詳細に紹介する。

2.3 深層学習

本節では、深層学習（ディープラーニング）における基本知識について説明する。機械学習はデータからの学習によって機械自身の特徴や傾向をつかみ、法則性を見つけ出してタスクを遂行する。データの特性から適切なニューラルネットワークモデルを構築する必要がある。これらのネットワークモデルの構築方法とパラメータ設定方法や学習データの必要性について説明する。

2.3.1 AI と機械学習

人工知能(Artificial Intelligence, 以下 AI と表記)は学習により認識能力を獲得し、画像認識、音声認識などを通じて問題解決や知識表現に展開している。すなわち人工知能とは典型的な人間の知的行動を機械に模造させることと言える。その要素技術は情報工学や数学、統計など様々な分野から得られている。AI は確立した学術的な定義や合意をされておらず、人が知的と感じること全般を広く指す[33]。機械学習は AI の一部であり機械に大量のデータを学習させて統計解析によって法則性を見つけ出しタスクを遂行させることを目的とする[34]。ディープラーニングは機械学習の手法の 1 つでありニューラルネットワークを用いたものを指す。機械学習には大きく 3 つのカテゴリ「教師あり学習(supervised learning)」、
「教師なし学習(unsupervised learning)」、
「強化学習(reinforcement learning)」がある。

- 教師あり学習：機械に入力データとそれに対する望ましい出力（答え）を与えることで、将来の未知のデータに対して予測ができるよう学習させるものである。分類(classification)や回帰(regression)の問題に利用される。
- 教師なし学習：機械に入力データのみを与え、外部からの指示なしに意味のある構造（クラスター）を自ら見つけさせるものである。クラスタリング(clustering)、次元削減(dimensionality reduction)、異常検知(anomaly detection)の問題に利用される。
- 強化学習：機械にエージェントとして環境の中で行動させることによって、報酬を得られる行動が何かを学習させるというものである。最適化問題(optimization problem)や遺伝的アルゴリズム(genetic algorithm)の問題に利用される。

本論文では教師あり学習を対象とする。教師あり学習では、対象物の特徴量を抽出するために、学習させる望ましい出力（ここでは測定値）を教師データとしてあらかじめ大量に準備する必要がある。多様な状態が考えられる場合に教師データの傾向に偏りがあると、未知のデータに対して推定結果のロバスト性に課題が残る。実産業分野における課題解決に本

研究を適用するには、質、量ともに適切な教師データをいかに準備するかが課題である。本論文では、過去に学習した知識モデルを活用して機械学習の効率を高めることと併用して、統計的手法を用いて多様な状態に対応した教師データを構築する方法を提案する。

2.3.2 ニューラルネットワーク

(1) パーセプトロン

人工ニューラルネットワーク(Artificial Neural Networks: ANN)あるいはニューラルネットワーク(Neural Networks: NN)は機械学習の一種であり、哺乳類の脳を構成する生物学的神経回路網の研究から影響を受けたコンピューティングシステムである。ニューロン(neuron)とは生物の脳を構成する神経細胞のことで、情報処理と伝播能力において特に優れている。人間の脳皮質では約 140 億ものニューロンが 3 次元で回路網を形成しニューロン間で情報を伝達することで、思考、認識などの情報処理を行うことができる。基本的な脳の構造は図 2-5 左のような構成[35]である。細胞体(soma)、樹状突起(dendrite)、軸索(axon)の 3 つの部分に分けられる。細胞体は細胞の中央部分にあたり、細胞核(nucleus)は細胞体の中に存在する。ニューロンが他のニューロンから電気信号により入力を受け取ると自身の中で電気を加算し、ある閾値を超えると次のニューロンにまた電気信号を送る。閾値の超過と関連ニューロンへの電気信号の伝達のことを発火(firing)という。各ニューロンは入力・出力ともに複数のニューロンと接続され、この接続をシナプス(synapse)という。各ニューロン間の接続強度は組み合わせにより異なるためネットワーク全体での電気信号の伝わり方が変わる。この違いによって人間は異なるパターンを認識できる。ニューラルネットワークは図 2-5 右のようにニューロンをノード(node)、シナプスに相当するノード間結合をエッジ(edge)として模擬した数理モデルで表現される。同モデルはパーセプトロン(perceptron)と呼ばれ、入力層と出力層の 2 層のネットワークを表す。

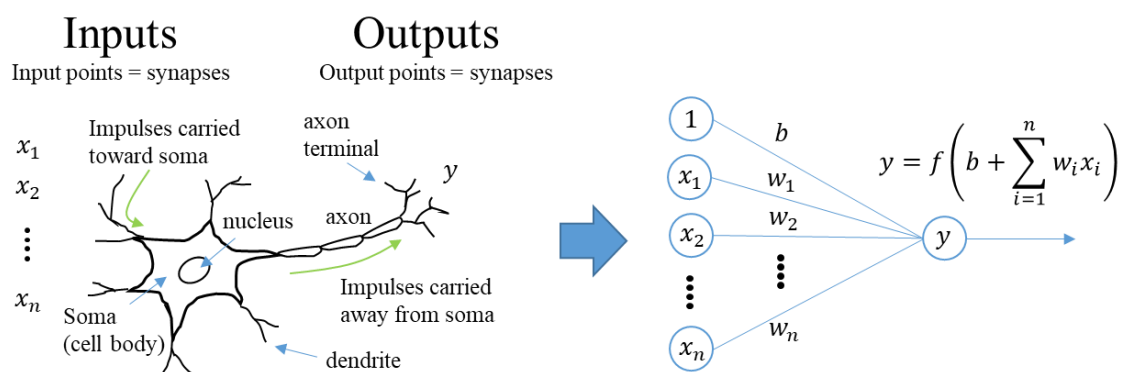


図 2-5 神経回路網の数理モデル

Fig.2-5 A mathematical model of artificial neurons that expresses neurons in the human brain and their connections.

パーセプトロンの入力の特徴量 n 個の値 (x_1, x_2, \dots, x_n) を持つ入力ベクトル x 、出力は 1 (発火する) あるいは 0 (発火しない) である。エッジには通過する値に乗じる重みベクトル w と、入力値を一定の範囲に偏らせるためにバイアス b を与えると式(2.18)で定義される。この $f(x)$ のことをステップ関数(Step function)と呼ぶ。

$$y = f\left(b + \sum_{i=1}^n w_i x_i\right) \quad (2.18)$$

$$f(x) = \begin{cases} 1 & (x > 0) \\ 0 & (x \leq 0) \end{cases}$$

ステップ関数のようにニューロンの線形結合後の出力に対して非線形変換を行う関数のことを活性化関数 σ (Activation function) と呼び、パーセプトロンではステップ関数が用いられる。ニューラルネットワークでは、活性化関数には、確率を出力する関数として任意の実数を 0 から 1 までの範囲に写像するシグモイド関数(Sigmoid function)や、正規化線形関数(Rectified linear unit, 以下 ReLU と表記)がよく用いられる。シグモイド関数は式(2.19)、ReLU は式(2.20)で定義される。

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2.19)$$

$$\sigma(x) = \max(0, x) \quad (2.20)$$

(2) ネットワークの学習

学習とは教師データ (トレーニングデータ) を用意し、ネットワークが出力 (予測値) とトレーニングデータとの誤差を最小化するように重みとバイアスの値を最適化し、予測モデルを生成することである。学習における主要なタスクには分類と回帰がある。分類は 1 あるいは 0 を出力する二値分類や複数の値を分類する多値分類があり、良品・不良品の判定、文書や画像のカテゴリといった離散値を出力する。回帰は、株価、気温や位置・状態情報などの連続値を出力する。回帰モデルの最終層は全結合層である。単一値で非線形の変換を行わずに、その一つ前の層から伝播された値の合計値を出力する。ニューラルネットワークにおける損失とは推定値と実測値の差 (誤差) を表す。回帰モデルの損失関数(loss function)には平均二乗誤差 MSE(mean squared error, 以下 MSE と表記) が一般的に用いられる。要素数を n 、 i 番目の実測値を y_i 、推定値を \hat{y}_i とすると MSE は式(2.21)で表される。

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.21)$$

ニューラルネットワークモデルの学習法の一つに勾配降下法(gradient descent)がある。図 2-6 に変数 w の損失関数 $L(w)$ のグラフを示す。勾配降下法はこの関数 L について w_0 から

スタートし、少しずつ移動しながら谷底の最小値 L_{min} を目指す。各ステップ γ における傾きは増加方向の最大値である。ステップ γ で到達した w_γ で評価される偏微分値に対して、反対方向をとることで谷側の方に移動できる。

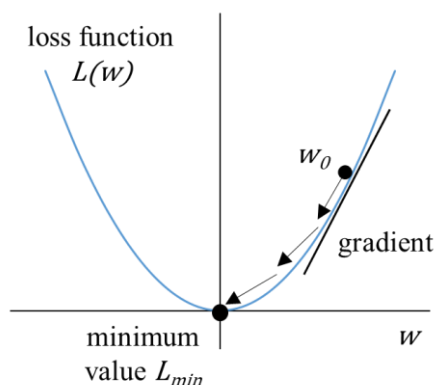


図 2-6 勾配降下法

Fig.2-6 An image of gradient descent.

パラメータを重み w とバイアス b , 損失関数を $L(w, b)$ とすると勾配降下法は式(2.22)のように数式で表される。

$$w_{k+1} = w_k - \eta \frac{\partial L(w, b)}{\partial w} \quad (2.22)$$

$$b_{k+1} = b_k - \eta \frac{\partial L(w, b)}{\partial b}$$

$\eta (> 0)$ は学習率(learning rate)と呼ばれる。パラメータの収束しやすさを調整する変数を表す。 η が小さいと学習が遅くなり、大きすぎると谷底を見失ってしまう。

シグモイド関数の式(2.19)の偏微分と ReLU の式(2.20)の偏微分をそれぞれ式(2.23), (2.24)に表す。

$$\frac{\partial \sigma(x)}{\partial x} = \sigma(x)(1 - \sigma(x)) \quad (2.23)$$

$$\frac{\partial \sigma(x)}{\partial x} = \begin{cases} 1 & (x > 0) \\ 0 & (x \leq 0) \end{cases} \quad (2.24)$$

要素数 n が大きい場合に計算時間やメモリ容量が膨大になるため、一般的にモデルの学習にはデータをランダムに 1 つずつ選択してパラメータを更新する確率的勾配降下法(stochastic gradient descent, 以下 SGD と表記)が用いられる。確率的勾配降下法は n 個のデータに対して式(2.25)で計算される。

$$\begin{aligned}w_{k+1} &= w_k - \eta(y_n - t_n)x_n \\b_{k+1} &= b_k - \eta(y_n - t_n)\end{aligned}\tag{2.25}$$

モデルを学習させる反復回数をエポック(epoch)と呼ぶ。エポック毎にデータをランダムにシャッフルすることにより学習に偏りが生じにくくなり、より最適解を得やすくなる。

(3) ハイパーパラメータ設定

本論文では SGD に設定する学習率に関しての最適化手法として Adam(Adaptive moment estimation, 以下 Adam と表記)[36]を使用する。Adam は SGD が持つ加速度成分に加えて運動量の概念を取り入れることでコスト計算の収束を高速化する。最適化するパラメータを θ 、学習率を α 、損失関数の勾配を g 、最適化に用いる損失関数を $f(\theta)$ とし、時間ステップ t における勾配を \hat{m}_t, \hat{v}_t とすると Adam の更新式は、勾配の 2 乗の移動平均 $v_t = E[g^2]$ 、勾配の移動平均 $m_t = E[g]$ を用いて以下の式(2.26)~(2.31)で計算される。

$$g_t = \nabla_{\theta} f_t(\theta_{t-1})\tag{2.26}$$

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t\tag{2.27}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2\tag{2.28}$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}\tag{2.29}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}\tag{2.30}$$

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t\tag{2.31}$$

ここで $\alpha, \beta_1, \beta_2, \epsilon$ はユーザから設定されるハイパーパラメータであり、本論文では $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-7}$ に設定する[35]。また最適化アルゴリズムが重みを更新する際の使用するデータサイズをバッチサイズ(batch size)と呼ぶ。エポック毎に検証データの判定における損失値を確認する。本論文では、5 エポック毎に学習率を 0.5 倍にするよう設定する[35]。

(4) 過学習

学習はコンピュータがデータからパターンや規則性を見つける作業を指す。学習により得られた予測モデルを用いて機械が見つけたパターンを未知のデータに当てはめて予測する。学習するデータの持つ情報の量に比べて、過剰に複雑なモデルを作ってしまう、外れ値やノイズまで学習してしまう現象を過学習(overfitting)という。モデルの複雑さは説明変数の数と説明変数それぞれの係数で決定される。説明変数の数が多い場合や、偏回帰係数の値における大小の差が著しい場合にモデルが複雑になってしまう。

$$y = a_1x_1 + a_2x_2 + \dots + a_nx_n + b$$

n 個の入力データに対して y は目的変数, a_n は偏回帰係数, x_n は説明変数, b は定数項とする. 過学習への対策には, 学習データを増やすこと, ハイパーパラメータを調整すること, および正則化がある. さらにネットワークモデルのノード情報の一部を隠すドロップアウトと呼ばれる方法がある.

正則化には, 必要のない説明変数を削除するために偏回帰係数を 0 にする (次元圧縮) L1 正則化と, 偏回帰係数を調整することでモデルを改善する L2 正則化がある. ここで損失関数を $f(x)$, パラメータ λ とすると次式で表される.

- 正則化なし: 損失関数

$$\min f(x)$$

- L1 正則化: 損失関数 + L1 正則化項

$$\min f(x) + \lambda \sum_{i=1}^n |w_i|$$

- L2 正則化: 損失関数 + L2 正則化項

$$\min f(x) + \lambda \sum_{i=1}^n |w_i|^2$$

(5) バッチノーマライゼーション

2015 年にバッチノーマライゼーション (Batch Normalization, 以下 BN と表記) 手法 [54] が提案された. 隠れ層の活性化関数の処理後の出力データ (アクティベーション) の分布に着目する. アクティベーションが 0 と 1 に偏った分布の時, 活性化関数がシグモイド関数の場合に S 字カーブの関数なので 0 または 1 に近づくとつれてその微分値 (傾き) は 0 に近づく. この場合には逆伝播での勾配の値が次第に小さくなり消えてしまう. これが勾配消失 (gradient vanishing) と呼ばれ, 層の深いネットワークでは勾配消失が深刻な問題になる. アクティベーションに偏りがある場合, 複数のニューロンがほとんど同じ値を出力するなら, 1 つのニューロンと同じ意味になる. これは表現力が制限される問題として学習がうまくいかない場合が起こる. ある層のパラメータ更新によって, その次の層への入力分布がバッチごとに大きく変化する内部共変量シフト (Internal covariate shift) が発生し, 学習が効率的に進まない問題が起こる. 学習効率に悪影響を及ぼす内部共変量シフトを抑えるために, すべての畳み込み後に特徴データを正規化する BN を用いて学習の収束速度を上げる. 勾配消失・爆発を防ぐための手法であり, 過学習を抑制する効果もある. 数式で表すと式 (2,32) [37] になる.

$$\begin{aligned}
\mu_B &\leftarrow \frac{1}{m} \sum_{i=1}^m x_i && // \text{ mini - batch mean} \\
\sigma_B &\leftarrow \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2 && // \text{ mini - batch variance} \\
\hat{x}_i &\leftarrow \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \varepsilon}} && // \text{ normalize} \\
y_i &\leftarrow \gamma \hat{x}_i + \beta \equiv \text{BN}_{\gamma, \beta}(x_i) && // \text{ scale and shift}
\end{aligned} \tag{2.32}$$

m 個の入力データの集合ミニバッチ $B = \{x_1, x_2, \dots, x_m\}$ に対して平均 μ_B , 分散 σ_B^2 を求め, 1 ミニバッチ内の全データ分布が平均 0, 分散 1 になるように正規化され $\{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_m\}$ に変換する. ここでプログラム実装において 0 で除算されることを防ぐため, 小さい値 $\varepsilon = 10^{-7}$ を分母に加える. γ と β はパラメータで初期値 $\gamma=1, \beta=0$ から学習によって適正值に調整される[37]. BN を活性化関数の前に入れることで, データの分布の偏りを減らすことができる. 正規化されたデータは固有のスケールとシフトで変換される.

2.3.3 ディープラーニング

パーセプトロンに対して, 多数の層を持つネットワークのことを多層パーセプトロン (Multilayer perceptron: MLP) と呼ぶ. 図 2-7 に入力層 (Input) が 1 層, 中間層 (Hidden) が 1 層, 出力層 (Output) が 1 層の一般的なニューラルネットワークを示す. 各ノードが最初の層から入力を受け取り, ある閾値を超えると次のニューロンにまた電気信号を送る. 1 層目から出力される情報は 2 層目の入力となり, 順に最終層まで伝達される. ディープラーニング (Deep learning) とは, 複数の深い階層構造を数理モデル化した複数の中間層を持つニューラルネットワークのことを指す.

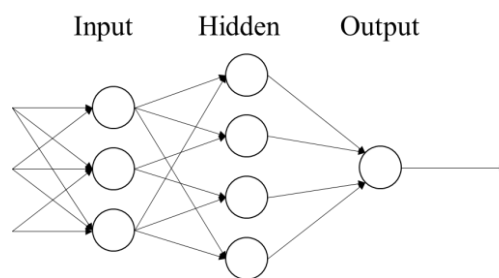


図 2-7 ニューラルネットワーク

Fig.2-7 A sample of a neural networks (multilayer perceptron).

ディープラーニングで広く活用されている代表的な技術には, 画像認識 (Image Recognition), 自然言語処理 (Natural Language Processing) の分野があげられ, 近年最も期待されている技術である. 画像認識分野では, 学習した画像のデータをもとに画像内の特徴を分析し, その特徴や法則性を見つけ出し, 分類することができる. 自然言語処理分野では, 会話や文章のやり取りなどにおいて, 人が普段使用している言語 (自然言語) を機械で

処理し、その言葉の持つ内容を抽出し、意味を解析することができる。

(1) 画像認識

画像認識(Image recognition)とは、画像や動画から文字や顔などのオブジェクトや特徴(形状、寸法、数、明暗、色など)を抽出(識別)して検出するパターン認識技術の一つである。画像認識技術は、顔認証や指紋認証などの生体認証システム、文字認識(OCR)、自動車の自動走行における障害物認識、工場の不良品検査など、幅広い分野ですでに広く利用されており、コンピュータビジョンなどの関連技術で重要な研究対象となっている。入力された画像から特徴量を抽出して分類するもので、画像分類、物体検出、異常検知、画像生成などのタスクで用いられる。本提案手法に関連するものでは、特に画像分類と物体検出があげられる。

- 画像分類(Image classification)は、対象とする画像があらかじめ設定されたカテゴリのどれに属するかを判断するタスクである。手書き文字認識のように画像全体のクラス分類は物体認識とも呼ばれる。
- 物体検出(Object detection)は、対象とする画像の中から特定の部品や欠陥箇所の位置を特定するタスクである。画像の中から対象物の位置までを特定する。

本論文は物体検出を対象とする。物体検出にはセグメンテーション(Segmentation)技術を用いる。セグメンテーションとは画像の中のピクセルレベルの情報に対してクラス分類を行う問題である。すべてのピクセルに対してピクセル単位で分類することで画像をオブジェクトごとに分割することができる。セグメンテーションの中には大きく 2 つのタスクがある。

- セマンティックセグメンテーション(Semantic Segmentation)と呼ばれ、入力画像のすべてのピクセルに対してピクセル単位でオブジェクトごとに色分けしてクラス分類する。セマンティックとはデータの構造と関連付けて意味を持たせるものである。
- インスタンスセグメンテーション(Instance Segmentation)と呼ばれ、画像データ中のすべてのオブジェクトに対してクラスラベル ID を付与して分類する。各オブジェクトに一意的 ID を付与するため、例えば 1 枚の画像に複数の人が写っている場合には、それぞれの人を別々のオブジェクトと認識して ID を付与する。セマンティックセグメンテーションとの主な違いは、空や道路のような定まった形を持たないオブジェクトには ID を付与しないため、空隙抽出には適さない。

本論文では、ピクセル単位で扱うセマンティックセグメンテーション技術を利用する。空隙情報をピクセル単位でラベリングする二値分類である。同技術は 2 つのカテゴリに限らず、対象物ごとのクラスにラベリングすることができる。これに対応する技術としてディープラーニングの代表的なアルゴリズムである畳み込みニューラルネットワーク(CNN)が使用される。近年例えば自動車の自動運転において、画像に対して物体の境界線を抽出して物

体を検出するセマンティックセグメンテーション技術やラベリング技術を用いて物体検出し、制御に活用するといった重要な役割を担う。物体検出の精度向上は、コンピュータによる物体の位置の理解、形状の理解に繋がり、様々な分野の研究への応用に期待されている。本論文の第3章では物体検出を対象とし、SEM画像内の空隙の場所を抽出する。この時のデータ構造を図2-8(a)に示す。このようなZ軸を使わないデータ構造において深層学習を適用するにはCNNが最適であり、これを利用する。

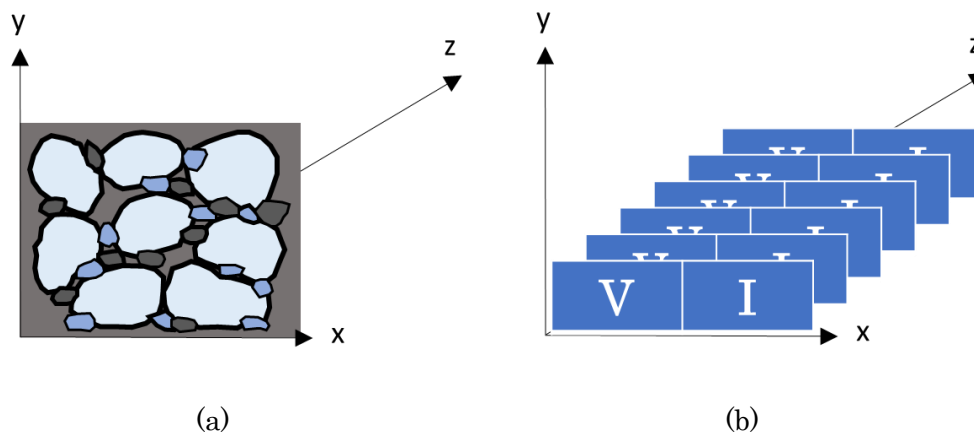


図 2-8 データ構造
Fig.2-8 Data structure.

(2) 自然言語処理

文章中の単語は記号の集まりとして表現できるが、記号をベクトルに表現することで、ベクトル間の距離や関係を計算して数値表現し、単語の意味を表現するモデルを使用する。これをベクトル空間モデル(Vector Space Models), あるいは単語の意味をベクトル空間の中で表現することから単語埋め込みモデル(Word Embedding Models)と呼ばれる。自然言語処理では、入力された自然言語から形態素解析、構文解析、意味解析、文脈解析の技術を用いて機械言語へ変換され、データとして利用することが可能になる。

- 形態素解析：それぞれの意味を担う最小単位（形態素）に文章を分割し、それぞれに品詞など各種情報を振り分けるタスクである。この作業によって文章の中にある形態素の意味をデータとして抽出することが可能になる。主に形態素解析エンジンを使って解析する。代表的なものには、無償では MeCab, ChaSen, KyTea, 商用では Rosette, IBM Watson, MARIMO などがある。
- 構文解析：形態素解析で分割した単語同士の関連性を解析し、主に文節間の依存構造を発見しツリー化するタスクである。依存構造解析と句構造解析などがある。
- 意味解析：辞書に基づいた意味を利用して正しい文を解析するタスクである。事前に用意するリストから文章中の固有名を結び付けてその意味を認識する。
- 文脈解析：複数の文章に形態素解析と意味解析を行い、文同士の関係性を解析するタ

スクである。

これに対応する技術としてディープラーニングのもう一つの代表的なアルゴリズムであるリカレントニューラルネットワーク(RNN)が使用される。ニューラルネットワークに時間の概念を取り込み、時系列データの扱いに特化したモデルをRNNと呼ぶ。時系列データの扱いについては特に自然言語処理分野を中心にモデルが構築された。LIBの場合に扱う充放電データは時系列データであるため、再帰型のRNNが用いられる。本論文の第4章では、時系列データである端子電圧や電流の測定値を用いてLIBのSOCを推定する。この時のデータ構造を図2-8(b)に示す。このようなデータ構造において深層学習を適用するにはZ軸を時間 t として扱う場合にRNNが最適であり、これを利用する。

2.3.4 CNNによる画像認識

近年、CNNがクラス分類において最も精度が良いとされ、注目されている。2014年にコンピュータによる物体認識の精度を競う国際コンテストImageNet Large Scale Visual Recognition Challenge(ILSVRC)コンペティションでの1000種類のクラス分類において、文献[38]に記載される手法がこれまでの手法に大差をつけて1位となり、画像認識分野で広く知られるようになった。CNNは画像内の空間的な特徴としての隣り合うピクセル同士の関連をうまくとらえることができる。

(1) CNNモデル

CNNは「畳み込み層」により特徴抽出する役割と「プーリング層」によりロバスト性を与える役割を持たせ、ニューロン同士の結合を制限する[39][40]。図2-9にCNN構造の概略を示す。

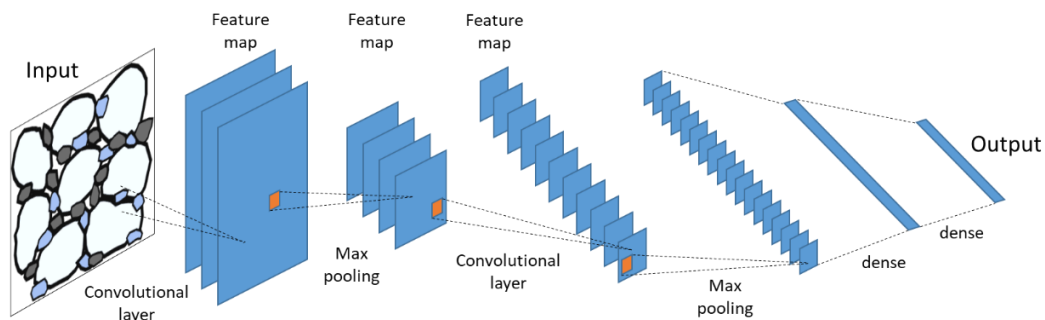


図 2-9 CNN 構造の概略図

Fig.2-9 An image of CNN structure.

畳み込みにはフィルタを用いて画像上でスライドさせ特徴抽出する。このフィルタの数をチャンネル数と呼ぶ。チャンネル数の方向に立体的に畳み込む。CNNは単純なクラス分類精度の良さだけでなく、認識を一方向に一層ずつ階層を進める際に、正規化線形ユニットと呼ばれる区分線形な活性化関数を使うことで、従来よりも高速化かつ最適化を図ることができる。CNNを用いた画像解析により、単一のピクセルの値のみでなく、ピクセルの位置

関係を考慮しながら特徴抽出できるため、本論文では、CNN を採用する。

(2) 全層畳み込みニューラルネットワーク (FCN) モデル

全層畳み込みニューラルネットワーク (FCN : Fully Convolutional Networks) モデル [41][42] は CNN の一種であり、すべてが畳み込み層で構成されるニューラルネットワークでセマンティックセグメンテーションを実現する。CNN が画像全体に対してクラスを出力するのに対し、FCN は画像と同サイズの行列を出力し、その行列の各要素に対応するピクセル毎のクラス分け結果を格納する。FCN を用いた LIB の空隙抽出では、この行列の情報を元に空隙と判別した箇所とそうでない箇所を色分けすることができる。しかし単純にプーリング層による逆畳み込み操作しか行っていないことに起因して、対象物の構造が欠損したり平滑化したりする欠点を持つ。そこで FCN が一連のプーリング層によりデコードするのに対し、High accuracy FCN モデル [43] では、CNN のコンボリューション層におけるエンコード段階の左右対称の Encoder - Decoder 構造で設定されたデコンボリューション層におけるデコード段階でアンプーリング層がその欠点を補い、より高精度な空隙抽出を実現する。

(3) U-Net モデル

U-Net モデル [44] は FCN の一種であり、ピクセル毎のクラス分けを行い、物体をクラスごとに色分けする。U-Net モデルのアーキテクチャを図 2-10 に示す。

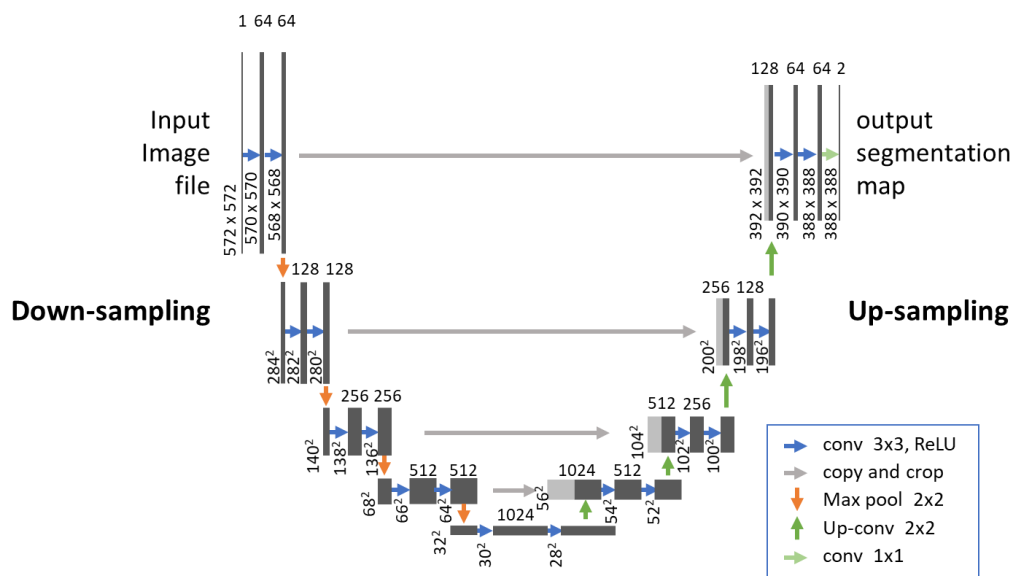


図 2-10 U-Net モデル [44]

Fig.2-10 U-Net model architecture.

High accuracy FCN と同様に左右対称の Encoder - Decoder 構造を持つ。U-Net が一般的な High accuracy FCN と異なる点は、エンコード段階 (ダウンサンプリング) で畳み込まれた解像度の特徴マップをその対応したデコード部分へ伝達し、それをデコードする際

(アップサンプリング) にエンコードにより失われた解像度の特徴マップ情報を補いながらデコードしているところであり、これが U-Net 最大の特徴である。この処理によりエンコード段階で失われた情報をデコード段階に伝達することができるため、より高精度にピクセル単位での物体の分類が可能になる。フィルタ数は最大 512 まで増やして畳み込みとプーリングを繰り返しエンコードする。ダウンサンプリング時に画像のサイズが縦横それぞれ半分になる際にチャンネルの数を倍に増やす。アップサンプリング時に画像が小さくなるため、対称となるダウンサンプリングの特徴マップからアップサンプリングの画像の大きさに合わせて切り出した(cropped)画像を足し合わせる。畳み込み層は全部で 23 層のネットワーク構造を持つ。

本論文では、一般物体認識で構造がシンプルな FCN モデルや U-Net モデルを用いて物体抽出精度を評価する。材料開発における SEM 画像データの画像認識に適合するディープラーニングを応用するのに最適なネットワーク構造を発見することを目的とする。その他 U-Net 以外にも、車や人物などの一般物体認識を対象とした RefineNet モデル[45]や Pyramid Scene Parsing Network モデル[46]、あるいは医療画像を対象として U-Net の特徴マップをネスト構造に改良した U-Net++モデル[47]などが報告されているが、リチウムイオン電池の空隙領域抽出への適用を考えた場合、どの手法でもデータに適したネットワーク構造設計が必要であり、U-Net と同様の議論に帰着する[48]。

2.3.5 RNN による時系列データ処理

CNN は入力の空間的特徴をとらえることに利用される。画像データでは 2 次元 (高さ×幅) で適用し、動画データでは時間を加えた 3 次元 (高さ×幅×時間) で適用される。次に RNN は入力情報間の時間依存性を利用するニューラルネットワークの 1 つである。入力データとしてテキスト、音声、時系列データの推定に広く用いられるモデルである。図 2-11 に RNN 展開図を示す。同図左は RNN を示し、時刻 t でセルは $x(t)$ を入力し $y(t)$ を出力する。隠れ状態 $h(t)$ はフィードバックさせて次の時刻 $t + 1$ で使用する。従来のニューラルネットワークと同様に入力、出力、および隠れ状態に対応する 3 つの重み U, V, W を持つ。同右は時間単位 (タイムステップ) を 3 要素の時系列処理として 3 層の RNN 展開を示す。各時刻で異なる入力に同じ操作を行えるためタイムステップ内で重みを共有する。タイムステップ T のデータ ($x(1), x(2), \dots, x(T)$) を 1 つの入力データ群として時系列データ群として扱う。これらデータの並びの規則性やパターンを学習することで、未来の時系列データに対して状態を予測することができる。

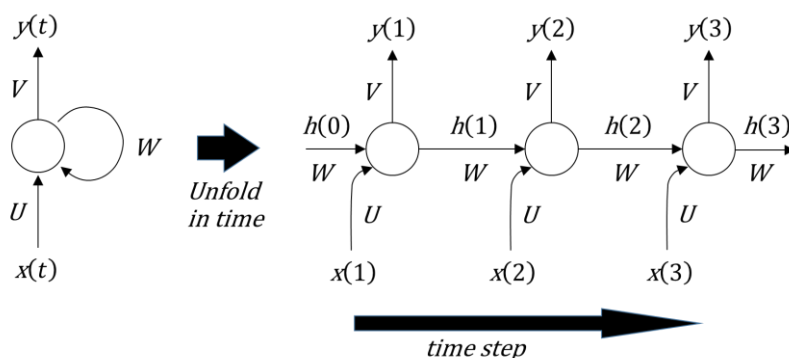


図 2-11 RNN 展開図

Fig.2-11 Architecture of RNN unfolded in time.

(1) LSTM モデル

RNN の中でも勾配消失・爆発対策に LSTM (Long Short-Term Memory, 以下 LSTM と表記) が広く適用される. LSTM[49]は最初に Hochreiter と Schmidhuber により提案され、今も手法の改良が行われている. 最近では LSTM を応用した手法として自然言語認識分野で GRU (Gated Recurrent Unit)[50], 双方向 RNN (Bidirectional RNN)[51], Attention RNN[52], QRNN(Quasi-Recurrent Neural Network)[53]など数多く報告されているが, LIB の SOC 推定への適用を考えた場合, どの手法でもデータに適したネットワーク構造が必要であり, 我々の提案する LSTM を利用した知識モデル活用法と同様の議論に帰着する.

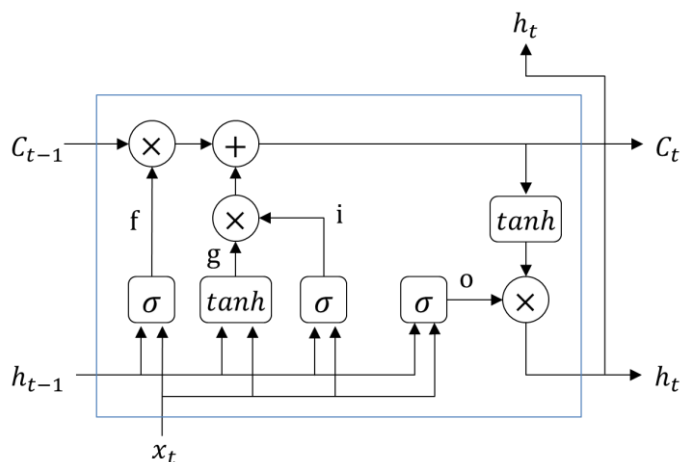


図 2-12 LSTM ブロックの内部構造[62]

Fig.2-12 Schematic setup of a LSTM block.

LSTM ネットワーク構造はニューラルネットワークとして入力層, 隠れ層, 出力層から成り, 隠れ層に長期依存性を学習するために LSTM ブロックと呼ばれる仕組みを構成する. 図 2-12 に LSTM の内部構造を示す[62]. LSTM ブロックの上線はセル状態 C で内部メモリを表し, 下線は隠れ状態 h を表す. 勾配消失問題を回避する仕組みとして出力ゲート o ,

忘却ゲート f , 入力ゲート i と内部隠れ状態 g を設定し, これらのパラメータを学習する. ゲートの活性化関数にシグモイド関数 σ と \tanh 関数を使用する. σ は式(2.33)で出力を $0 \sim 1$ の値に変換される. \tanh は式(2.34)で計算され, 入力値を $-1 \sim 1$ の範囲に正規化することで二次微分の減衰がゆっくりゼロになり, 勾配消失問題が起きにくくなる.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2.33)$$

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.34)$$

以下に具体的な計算方法を示す. まず出力ゲート o では, 入力 x_t と前の隠れ状態 h_{t-1} が o の直前の活性化関数 σ に入力する各エッジの重みを w_{xo} , w_{ho} とすると, o はそれぞれ重みをかけてバイアス b_o を加えた式(2.35)で計算される. 次に忘却ゲート f では, 同様に x_t と h_{t-1} が f の直前の活性化関数 σ に入力する各エッジの重みを w_{xf} , w_{hf} とすると, f は式(2.36)で計算される. 次に入力ゲート i では, 同様に直前の活性化関数 σ に入力する各エッジの重みを w_{xi} , w_{hi} とすると, i は式(2.37)で計算される. 次に内部隠れ状態 g では活性化関数に \tanh 関数 (Hyperbolic tangent function: 双曲線正接関数) を用いる. x_t と h_{t-1} が g の直前の \tanh 関数に入力する各エッジの重みを w_{xg} , w_{hg} とすると g は式(2.38)で計算される. 入力ゲート i の出力によって g の出力は変換される. 次にセル状態 C は長期的な依存関係を学習する長期記憶を担い, o, f, i, g が与えられると式(2.39)で計算される. 具体的なデータの流れは, 時刻 $t-1$ のセル状態 C_{t-1} に忘却ゲートの値を乗算し, 内部隠れ状態 g に入力ゲートの値を乗算した値へ加算して次の時間層に C_t を送ることで, 逆伝播時にも勾配が極端に変動しないようにして勾配消失問題を回避する. 次に隠れ状態 h は短期記憶を担い, o, C が与えられると式(2.35)と(2.39)から式(2.40)で計算される.

$$o_t = \sigma(w_{xo}x_t + w_{ho}h_{t-1} + b_o) \quad (2.35)$$

$$f_t = \sigma(w_{xf}x_t + w_{hf}h_{t-1} + b_f) \quad (2.36)$$

$$i_t = \sigma(w_{xi}x_t + w_{hi}h_{t-1} + b_i) \quad (2.37)$$

$$g_t = \tanh(w_{xg}x_t + w_{hg}h_{t-1} + b_g) \quad (2.38)$$

$$C_t = i_t \otimes g_t \oplus C_{t-1} \otimes f_t \quad (2.39)$$

$$h_t = o_t \otimes \tanh(C_t) \quad (2.40)$$

(2) モデルの学習

LSTM モデルに教師データを入力して学習させる. ニューラルネットワークの損失とは推定値と実測値の差であり, そのままの値を出力する回帰モデルを対象とするため, 損失関数 (評価関数) には平均二乗誤差 MSE(mean squared error, 以下 MSE と表記) を用いる. 要素数を n , i 番目の実測値を y_i , 推定値を \hat{y}_i とすると MSE は式(2.41)で表される. 学習では損失関数が最小値をとるようなパラメータを求める.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.41)$$

モデルの学習には確率的勾配降下法(stochastic gradient descent, 以下 SGD と表記)を用いる[62]. SGD に設定する学習率に関しての最適化手法として 2.3.2 節 (3) で説明した Adam を使用する. Adam は SGD が持つ加速度成分に加えて運動量の概念を含ませることでコスト計算の収束を高速にする.

(3) ハイパーパラメータ設定

LSTM ではその他にも推定精度に大きく影響を与える 3 つのハイパーパラメータ (データ分割長, 隠れノード数, エポック数) があり, 対象とする特徴空間やラベル空間にあった最適設定が必要である. 時系列データの扱いに特化した LSTM モデルにおいて, T 個のデータ $(x(1), x(2), \dots, x(T))$ といった 1 つの入力データ群として同データ群を複数扱う. これらデータの並びに規則性・パターンを学習することで, 未来の時系列データが与えられた時にその状態を予測することができる. 過去を遡る逆伝播における計算単位として時間依存性を見るデータ長を表すデータ分割長(maxlen) を設定する. データ長全体を一度に扱うのが理想的に見えるが, 現実にはデータ長を長くすると勾配消失 (爆発) 問題が起これり, いい結果が得られない. 次に隠れノード数(HIDDEN_SIZE) は LSTM 内部状態の深さを表す. またエポック数(EPOCHS) は対象とするデータ全体に対して繰り返し学習する回数を表す. 学習は必ずしもエポック数を増やす必要はなく回数と精度の関係性を確認して適切に設定する. 放電パターンや放電周期の違いによるこれらパラメータとの関係性を検証する.

2.3.6 転移学習

転移学習はある領域の知識を別の領域の学習に適用させるディープラーニング学習技術である. 転移学習の流れは, 人が過去の経験を活かして新しいスキルを習得するように, ある領域で学習した知識モデルを転移させることで学習効率を向上させることができる. しかし転移元と転移先の関連性が低い場合や転移方法が適していないと精度が悪化する場合があり, 使用には事前確認が必要である.

表 2-1 に転移学習の表記と定義を示す. 領域(ドメイン) D は特徴空間 X と周辺確率分布 $P(X)$ で構成される. タスク T はラベル空間 Y と条件付き確率分布 $P(Y|X)$ で構成される[54]. ソースは知識の転移元を指し, ターゲットは知識の転移先を指す. 知識の転移元のドメインはソースドメイン D_S , タスクはソースタスク T_S とする. 同様に, 知識の転移先のドメインはターゲットドメイン D_T , タスクはターゲットタスク T_T とする. 転移学習とは D_S, T_S, D_T, T_T が与えられた時, $D_S \neq D_T$ または $T_S \neq T_T$ において, D_S や T_S の情報を使って予測関数 η を学習することである. 転移学習は $D_S \neq D_T$ または $T_S \neq T_T$ のときに用いられる. $D = \{X, P(X)\}$, $T = \{Y, P(Y|X)\}$ の定義から, 転移学習の扱う問題は以下の 4 つに分けられる.

表 2-1 転移学習の表記と定義

Table 2-1 Notations and definitions of Transfer Learning

Domain component	Task component
$D = \{X, P(X)\}$	$T = \{Y, P(Y X)\}$
X : Feature space	Y : Label space
$X = \{x_1, \dots, x_n\}, x_i \in \mathcal{X}$	$Y = \{y_1, \dots, y_n\}, y_i \in \mathcal{Y}$
$P(X)$: Marginal probability distributions	$P(Y X)$: Conditional probability distributions
x_i : Feature vectors	

- 1) 特徴空間が異なる問題 ($X_S \neq X_T$)
- 2) 特徴空間は同じで周辺確率分布が異なる問題 ($P(X_S) \neq P(X_T)$)
- 3) ラベル空間が異なる問題 ($Y_S \neq Y_T$)
- 4) ラベル空間は同じで条件付確率分布が異なる問題 ($P(Y_S|X_S) \neq P(Y_T|X_T)$)

本論文で対象とする LIB の SOC 推定は、同じ電池仕様の知識と異なる電池仕様の知識の両方を扱い、異なる電池仕様の知識を転移元とする場合にも、2)の特徴空間は同じで周辺確率分布が異なる問題に該当することから、転移学習の問題として扱うことができる。この転移学習は同質な転移学習(Homogeneous Transfer Learning)と呼ばれる。どちらの電池仕様においても、同質な転移学習では、ソースドメインで事前に学習した部分的なネットワークをネットワーク構造や重みなどを含めて再利用することができる。転移学習のアプローチには、過去に学習した知識モデル(学習済みネットワーク構造と重み)を固定したまま、この後ろのレイヤに層を追加して新たな層の重みのみを学習する転移学習と、知識モデルを初期値として、モデル全体の重みを再学習させるファインチューニング(Fine Tuning)[55]がある。深層学習を活用する時には、膨大なトレーニングデータを用いて学習するために多くの学習時間を要する。この課題への対応策として転移学習の学習効果に期待される。過去の知識モデルを利用して学習効率を向上させ、予測モデル作成に要する学習時間の短縮が可能になることに期待が高い。

2.4 まとめ

本章では、まず LIB の構造と等価回路モデルについて説明した。LIB 材料開発において構成する粒子や活物質、導電助剤やバインダーなどの空間的内部構造が電池特性を向上するのに重要な要因であることを説明した。電解液に相当する空隙構造が、リチウムイオンの通りやすさ、導電助剤の効果などで電池特性に影響し、電池性能を左右する重要な情報となることを説明した。

次に、モデルベースの状態推定法として、この等価回路モデルに基づいた、従来の研究に

における蓄電池の SOC 推定方法について説明した。蓄電池の SOC 推定方法はこれまでに数多く提案されている。中でも最も一般的に使われている電流積算法とカルマンフィルタ (EKF)法について紹介した。各推定方法に対して、正しく使用すること、最適な条件下で使用することで、精度の高い推定ができることを紹介した、しかしながら、電流積算法は補正機能がないために初期値誤差やセンサ誤差が蓄積することで高精度化が難しいこと、EKF 法はノイズ環境設定について最適値が変動することから、推定精度に大きく影響することといった課題が残ることを説明した。

次に、対象物の物体検出や蓄電池の SOC 推定に深層学習を利用するための必要な要素技術について説明した。まず AI や機械学習における深層学習 (ディープラーニング) の位置づけを紹介した。ディープラーニングで広く活用されている代表的な技術には、画像認識 (Image Recognition)、自然言語処理 (Natural Language Processing) の分野があげられ、近年最も期待されている技術である。画像認識は、学習した画像のデータをもとに画像内の特徴を分析し、その特徴や法則性を見つけ出し、分類することができる技術であり、本論文の物体抽出で使用する CNN 技術を紹介した。自然言語処理分野は、会話や文章のやり取りなど人が普段使用している言語からその言葉の持つ内容を抽出し、意味を解析することができる技術であり、隣接するデータとの関連性を時系列で特徴や法則性を見つけ出すことができる。本論文の時系列データを用いた SOC 推定で使用する RNN 技術を紹介した。

深層学習を利用した応用研究では、主にディープラーニングのネットワーク構成とパラメータ設定方法を対象分野にどのように適応させるかということが求められる。さらにディープラーニングには大量の教師データが必要になるため、適切なデータをどのように準備するのか、過去の知識モデルを活用するなど、適切な質、量を持つ教師データを準備する必要がある。さらにディープラーニングの学習は多くかかり、ハードウェアアクセラレータ GPU を利用して高速化が取り組まれているが、リアルタイム処理できる時間までには至らない。学習のオフライン処理と分類や回帰 (推定) のオンライン処理と組み合わせて求める課題に対処する必要がある。深層学習を活用するには、最適なパラメータ条件設定を含む「適切なネットワークモデルの構築」や、課題に対処するために適切な質、量の「教師データの準備」と、運用方法も含めた「学習時間の短縮」が大きな課題として認識される。

第3章 深層学習を用いた電池構造抽出法

本章では、深層学習において、画像の空間的な特徴をとらえることのできる CNN モデルを利用した電池構造抽出方法を説明する。

3.1 はじめに

電池材料開発において、人は新しい電池材料および試料配合などの組み合わせや手順を変えて設計条件を決めて繰り返し実験することによりその効果を確認して、顧客要件など目標とする特性値（機能、性能）を持つ新材料や代替材料を探索する。その際、注目する実験試料の分散具合や構成比、立体的構造を確認するために、実験結果の画像から注目する実験試料の特徴量を抽出するのに非常に時間を要している。しかも人の習熟度によって抽出精度にバラツキが生じて、材料開発を効率的に進めることができない。本手法を用いてこの作業を自動化し、材料開発スピードを加速させることを目的とする。

本論文は深層学習を活用した LIB の SEM 画像データ内の空隙情報を自動抽出する画像解析アルゴリズムを開発し、LIB の空隙抽出に適用した結果を報告する。本提案は 2 章で説明した MI の 2 つのアプローチのうち、(1)蓄積した LIB 画像データから空隙情報を自動抽出し数値化してデータベースに格納することを対象とする。画像データから LIB の空隙情報を自動抽出し数値化することおよび熟練者と同等の抽出精度を目標値として設定する。

3.2 深層学習を活用した電池構造抽出

本節では、ディープラーニングを活用した本提案手法を説明する。深層学習を利用するために、適切なネットワークモデルの構築と教師データの準備方法について説明する。

3.2.1 ネットワークモデルの構築

ネットワークモデルによる抽出精度を比較するために、以下の 3 つのネットワークモデルを適用し、空隙抽出用に用意した SEM 画像データを用いて LIB の空隙情報を抽出する。次に説明するネットワークモデルの図中に記載する数値(H,W,C)は画像サイズ(Height x Width)とチャンネル数 C を意味する。

(1) FCN モデル 1

High accuracy FCN モデルを使用する。図 3-1 に示すように、各層のフィルタ数を順に 2, 4, 8 個と畳み込み、次にサイズ 6x6 ヘプリーングを 16 個フィルタ数行い、さらに畳み込みとプリーングを繰り返しエンコードする。次に Encoder - Decoder 構造でアンプリーングとデコンボリューションによりデコードする。

(2) FCN モデル 2

High accuracy FCN モデルを使用する。図 3-2 に示すように、FCN モデル 1 とは違うフィルタ数やプリーング条件等各種パラメータ条件へ変更する。各層のフィルタ数を最大 128 まで増やして畳み込みとプリーングを繰り返しエンコードする。同様に Encoder - Decoder

構造でデコードする。

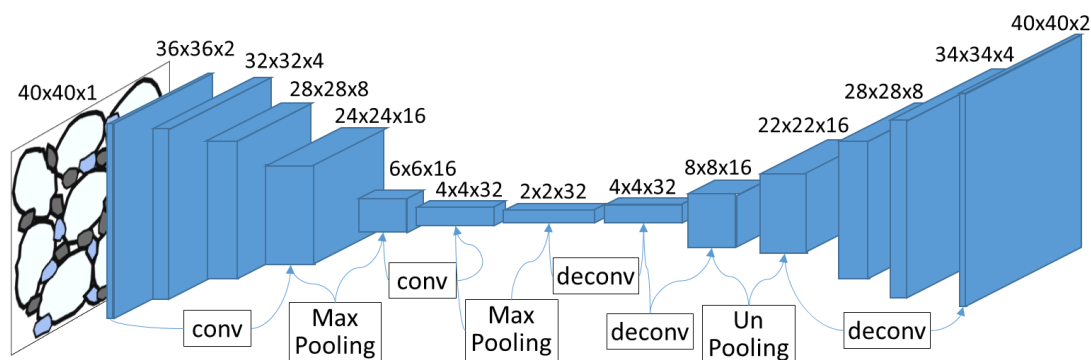


図 3-1 空隙抽出用 FCN モデル 1

Fig.3-1 FCN Model 1 for pores extraction.

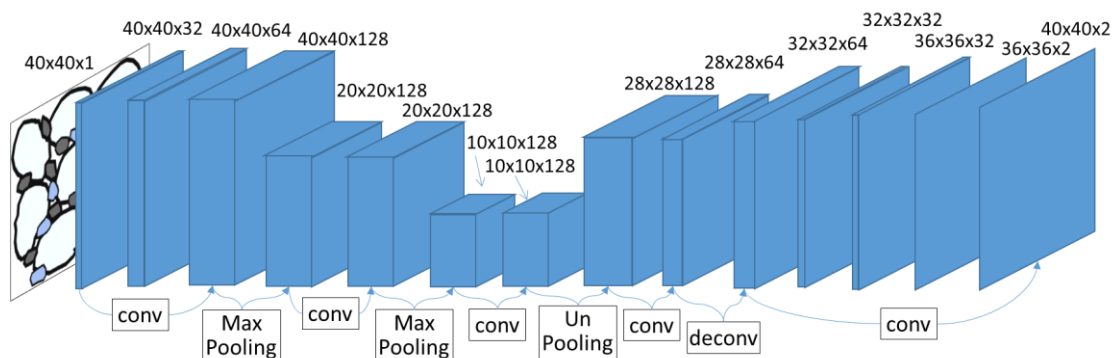


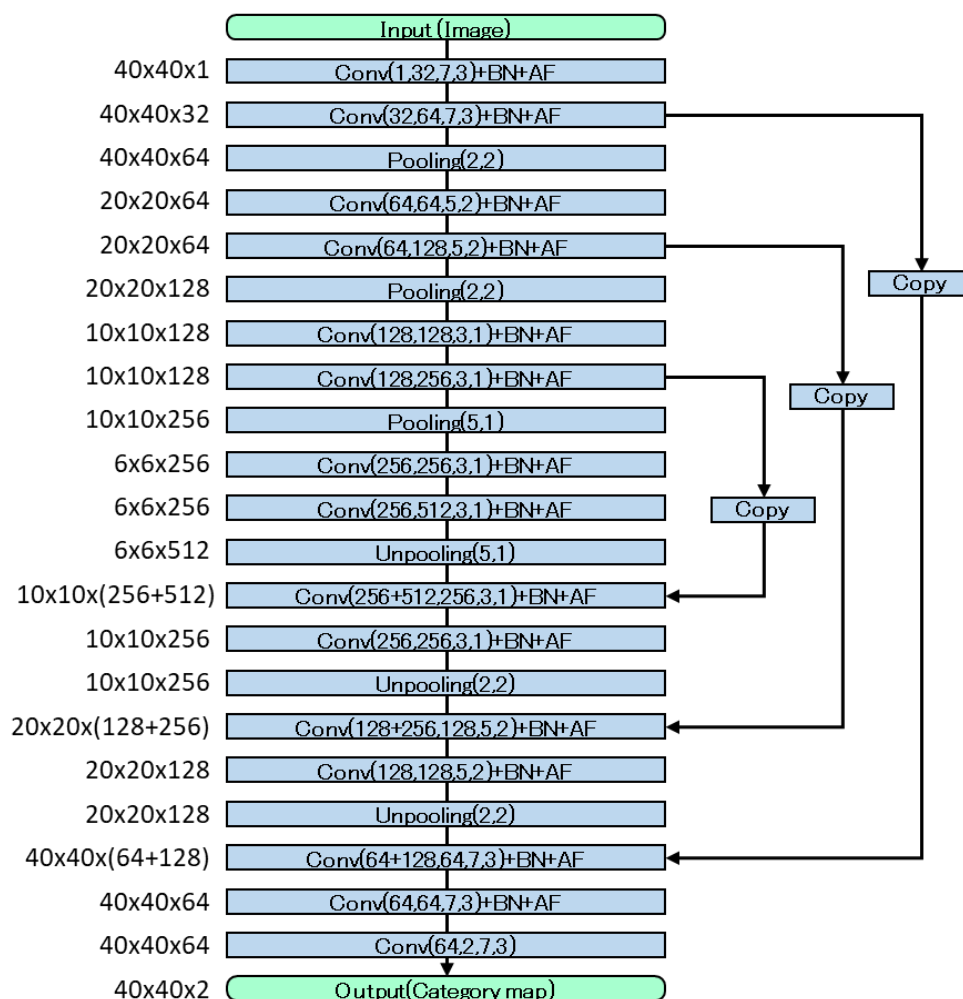
図 3-2 空隙抽出用 FCN モデル 2

Fig.3-2 FCN Model 2 for pores extraction.

(3) U-Net モデル

第 2 章 図 2-10 で示した U-Net モデルのネットワーク構造を進化させ、新規に独自のネットワークモデルを構築した。本提案ネットワークモデルの処理フローを図 3-3 に示す。フィルタ数は最大 512 まで増やして畳み込みとプーリングを繰り返しエンコードする。本提案モデルは U-Net モデルの一種だが、完全に左右対称な形の Encoder-Decoder 構造を持たせたことにより、エンコード部分の解像度の特徴マップ情報を切り出すことなくそのままコピーして対称となるデコード部分へ伝達し、エンコードにより失われた情報を補いながらデコードする。深いネットワークでは、ある層のパラメータ更新によって、その次の層への入力分布がバッチごとに大きく変化する内部共変量シフト(Internal covariate shift)が発生し、学習が効率的に進まない問題が起こる。学習効率に悪影響を及ぼす内部共変量シフトを抑えるために、すべての畳み込み後に特徴データを正規化するバッチノーマライゼーション(Batch Normalization, 以下 BN と表記)[54]を用いて学習の収束速度を上げる。BN は

1 ミニバッチ内の全データ分布が平均 0, 分散 1 になるように正規化する処理である. 勾配消失・爆発を防ぐための手法であり, 過学習を抑制する効果もある. さらにアップサンプリングでは畳み込みによる画像のぼかし効果やロバスト性向上のために, アンプーリングを用いて条件を最適化するなど, 独自のネットワークモデルを構築した. 尚, 活性化関数 (Activate Function : 以下 AF と表記)には ReLU(ランプ関数)を適用する.



Convolution: Conv(input channels, output channels, filtersize, padding), all strides 1
 Max pooling: Pooling(filtersize, strides)
 Batch Normalization: BN
 Activate Functions (ReLU): AF
 Copy Feature Map: Copy:

図 3-3 空隙抽出用の本提案ネットワークモデルの処理フロー

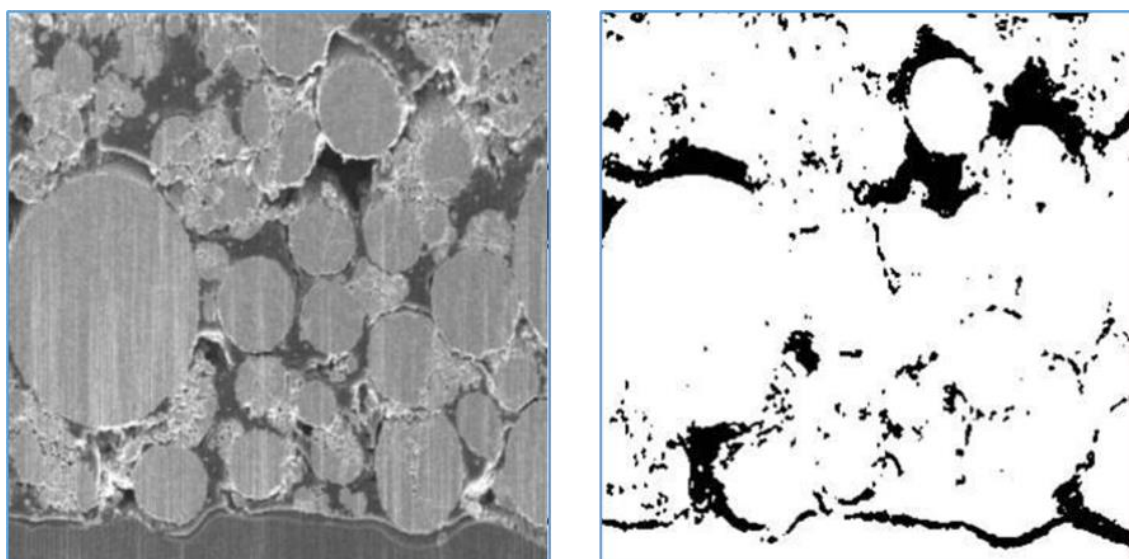
Fig.3-3 The flowchart of the proposed network model for pores extraction.

それぞれ 7x7, 5x5, 3x3 フィルタを用いて, すべて 1 ストライドで順に畳み込み(+それぞれ 3, 2, 1 のゼロパディング付加), その都度出力データに BN+ReLU 処理を行い, 2 回の 2x2, ストライド 2 のマックスプーリングで半分のコンポーネント数, 2 倍のチャンネル数

へ、1回の5x5、ストライド1のマックスプーリングでダウンサンプリングする。また、アップサンプリングには、3x3, 5x5, 7x7 フィルタで、すべて1ストライドによる畳み込み(+それぞれ1, 2, 3のゼロパディング付加)と、その都度出力データにBN+ReLU, を行い、1回の5x5, 1ストライドのアンプーリングと2回の2x2, 2ストライドのアンプーリングを行う。本手法では、アンプーリングの適用や、特徴マップをクロップせずそのままコピーする。最後の7x7 フィルタ+1ストライド+3ゼロパディングの畳み込みを使用して、各64コンポーネントの特徴ベクトルを目的のカテゴリにマッピングする。

3.2.2 教師データの準備

教師データの準備では、熟練者により人手でビットマップ上に空隙箇所をマーキングして、空隙の教師データ(真値)を作成した。図3-4にLIB画像において熟練者がマーキングした教師データの空隙領域の画像例を示す。(a)がSEM画像、(b)が人手で抽出した空隙情報(黒い領域)である。



(a) SEM 画像

(b) 人手で抽出した空隙情報(黒:空隙箇所)
(training data).

図3-4 LIB画像における学習用の空隙領域

Fig.3-4 A training image of pores in LIB.

3.2.3 提案手法

本論文では、深層学習を活用し畳み込みニューラルネットワーク(CNN)の中のU-Netモデルから新たな独自ネットワークモデル構造を構築し、同モデルを用いてSEM画像データから空隙情報を自動抽出する方法を提案する。本手法ではU-Netモデルネットワーク構造をもとに、学習効率に悪影響を及ぼす内部共変量シフト対策に加えてフィルタ設計などでロバスト性向上を図り、特徴マップを活用したアップサンプリング処理を加えた独自のネ

ネットワーク構造を提案する。具体的には 3.2 節 (3) U-Net モデルで詳細を説明した。また、教師データ構築に対して本手法の独自の方法として、データ拡張(ロバスト対策)機能を持つ。図 3-5 にロバスト処理モデルを示す。データ拡張の方法は、まず任意に指定した割合に応じて、それぞれトレーニングデータとテストデータの中からランダムにデータを選択し、次に選択されたデータを、コピーし回転した後、ぼかし、コントラスト変更、ノイズ添加のうち一つをランダムに処理するものである。本機能を用いることで教師データ数は FCN1, FCN2 と比較して約 2 倍のデータ数を準備することで、本手法のディープラーニングのロバスト性を向上させる。

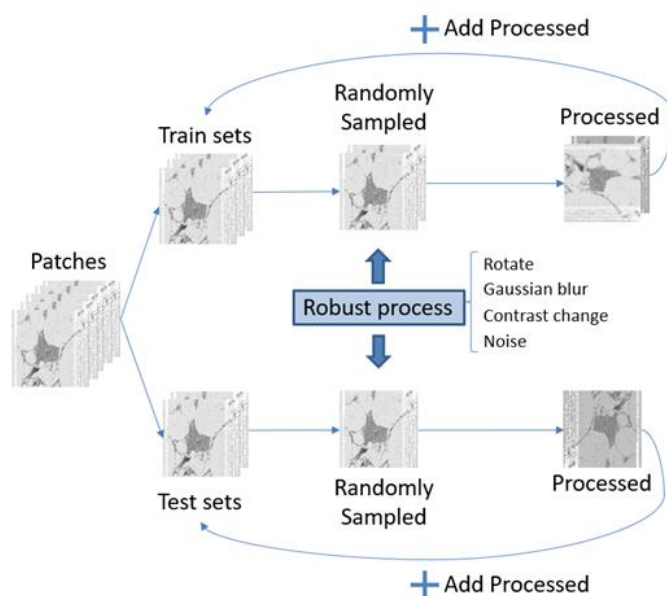


図 3-5 ロバスト処理モデル

Fig.3-5 The model of robust process.

3.3 評価指標

ネットワークモデルの有効性を評価するために、次の 5 つの評価指標を定義する。

- (a) 真陽性率(TPR)
- (b) 偽陽性率(FPR)
- (c) ROC 曲線と AUC
- (d) Pixel Accuracy
- (e) Intersection over Union(IoU)

3.3.1 真陽性率(TPR)と偽陽性率(FPR)

データが陽性(空隙である)か陰性(空隙でない)かを予測する二値分類器を評価するために、次の表 3-1 の概念を適用する。

表 3-1 対応表

Table 3-1 Confusion matrix.

		True 正解データ	
		Positive 陽性(空隙である)	Negative 陰性(空隙でない)
Predicted 予測結果	Positive 陽性(空隙である)	TP(True Positive) 真陽性	FP(False Positive) 偽陽性
	Negative 陰性(空隙でない)	FN(False Negative) 偽陰性	TN(True Negative) 真陰性

これらを用いると正解率 ACC は次の通りになる.

$$ACC = \frac{\text{正しく陽性・陰性を判定した数}(TP + TN)}{\text{ピクセル総数}(TP + FP + TN + FN)} \quad (3.1)$$

しかし ACC ではクラス間のデータ数に偏りがあると、データ数の多いクラスの結果が ACC に影響する。従って評価指標にはクラス間のデータ数の偏りによる影響を受けないよう、以下の式(3.2), (3.3), (3.4)を用いる。

- 真陽性率 (True Positive Rate, TPR, 感度)

$$TPR = \frac{\text{空隙であると正しく予測した数 } TP}{\text{実際に空隙である合計数 } (TP + FN)} \quad (3.2)$$

- 真陰性率 (True Negative Rate, TNR, 特異度)

$$TNR = \frac{\text{空隙でないと正しく予測した数 } TN}{\text{実際に空隙でない合計数 } (FP + TN)} \quad (3.3)$$

- 偽陽性率 (False Positive Rate, FPR, 1-TNR)

$$FPR = \frac{\text{空隙であると誤って予測した数 } FP}{\text{実際に空隙でない合計数 } (FP + TN)} \quad (3.4)$$

3.3.2 ROC 曲線と AUC

ROC(Receiver Operating Characteristic)曲線は機械学習のモデル評価等の精度の比較に用いられ、定量的な予測について、陽性(空隙である)、陰性(空隙でない)を予測する場合の判断基準として閾値(Cutoff point)を設けて分類する[53]。図 3-6 に予測分布のイメージ図を示す。一般的に陽性と陰性の予測分布は図 3-6 のように真ん中のところで重なってお

り閾値の位置で TPR, FPR は変動する。つまり閾値を高め設定すると偽陰性が少なくなるが偽陽性が増える。逆に低めにすると偽陽性が少なくなる。

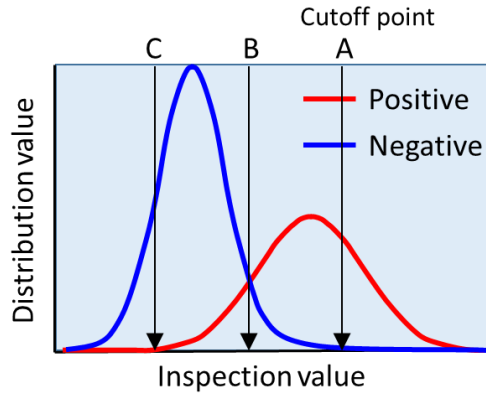


図 3-6 陽性と陰性の予測分布のイメージ図

Fig.3-6 The distribution image of Predicted Positive and Negative.

ROC 曲線は縦軸を TPR, 横軸を FPR とし, プロットして設定する。まず予測結果の分布図を用いて, どの値から陽性と判断するか閾値を決める。その閾値で陽性, 陰性として分類した数と正解データの陽性と陰性の合計数によって, 式(3.2), (3.4)を用いて TPR と FPR を計算する。同様に他の閾値を図 3-6 の(A→B→C)と変動させた場合の TPR と FPR の値をプロットし ROC 曲線を描く。図 3-7 に ROC 曲線の例を示す。

目指すべき閾値として, 正しく“空隙である”と判定する割合を大きく (TPR が大) することと, 間違っって“空隙である”と判定する割合を小さく (FPR が小) することが求められる。異なる予測手法の優劣を比較する場合には, ROC 曲線がより左上方に位置するほど性能が高い, すなわち ROC 曲線と x 軸 y 軸で囲まれた部分の面積 AUC(Area Under the Curve) が 1 に近いほど性能が高いと言える。

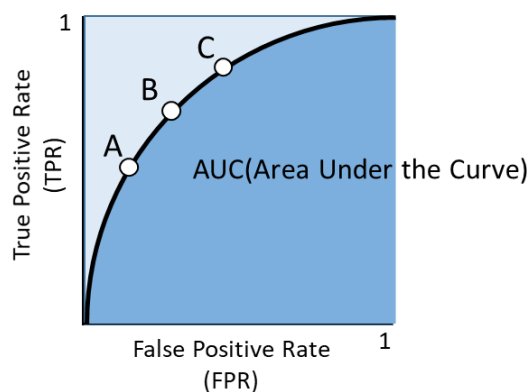


図 3-7 ROC 曲線の例

Fig.3-7 An example of ROC curve.

3.3.3 Intersection over Union (IoU)

機械学習のような二値(陽性, 陰性)分類器の概念で予測する手法に対して, クラス間のデータ数の偏りによる影響を受けないよう性能評価する指標の一つに **Intersection over Union**(以下, **IoU**)がある[54]. LIB 画像データ全体に占める空隙データの割合は小さいことが多く, ピクセルが「空隙でない」と予測して実際に空隙でない」の **TN** は大きな値を持つ. ゆえに **IoU** は **TN** 以外の, ピクセルが「予測結果が陽性」と「実際に陽性」の和集合 (**TP+FP+FN**)に対する「陽性と予測して実際に陽性」**TP** の割合で式(3.5)のように計算される. 物体認識において **IoU** が大きいとは空隙と予測した領域と実際に空隙だった領域の重なりが大きいことを表し, 1 (100%) に近いほど性能が高い.

$$IoU = \frac{TP}{TP + FP + FN} \quad (3.5)$$

3.3.4 Pixel Accuracy

二値分類器における画像の全ピクセルに対して, 予測結果と正解データが一致するピクセルの合計数 n とし, **Pixel Accuracy** は全ピクセル数 T に対する n の割合として算出され, 式(3.6)を用いる.

$$Pixel\ Accuracy = \frac{\text{正解と一致するピクセル合計数 } n}{\text{全ピクセル数 } T} \quad (3.6)$$

すべてのデータを用いて重みを更新した場合を 1 エポックと呼ぶ. データにはトレーニングデータとテストデータの 2 種類あり, それぞれの **Pixel Accuracy** が計算される. テストデータに対する予測結果について, **Pixel Accuracy** 値が収束する時のエポック数を最適なエポック数として設定する. 図 3-8 に **Pixel Accuracy** の計算例を示す.

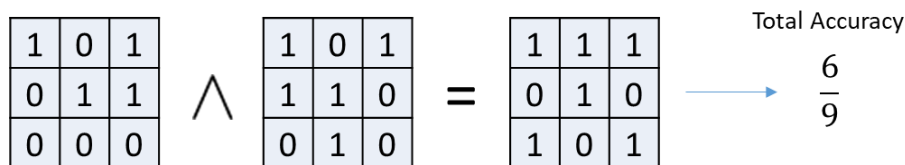


図 3-8 Pixel Accuracy 計算例

Fig.3-8 An example of Pixel Accuracy computation.

3.4 本手法の実験方法

本報告では, SEM 画像を解析して LIB の構成物質を抽出し数値化することを目的とし, 空隙を抽出対象とした.

3.4.1 開発環境

本実験に使用した開発環境のハードウェア, ソフトウェア構成は以下の通りである.

【ハードウェア】

- CPU : Intel®core™i5-6500CPU@3.20GHz
- GPU : NVIDIA GeForce GTX 1050 Ti
- RAM : 16GB, VRAM : 4GB

【ソフトウェア】

- OS : Windows10 Enterprise
- プログラミング言語 : Python 3.5
- ディープラーニングライブラリ : Chainer 3.2
- GPU プログラミングライブラリ : CUDA, CuPy, CuDNN
- その他の主なライブラリ : NumPy, openCV, Matplotlib
- IDE : PyCharm

本実験ではプログラム言語に Python を使用する。Python はシンプルで覚えやすいプログラム言語で、オープンソースであり無料で自由に利用することができる。Python は科学分野、特に機械学習やデータサイエンスの分野でよく使われている。NumPy や SciPy といった数値計算や統計処理を行うライブラリや、Matplotlib といったグラフ描画のためのライブラリや、TensorFlow や Chainer などのディープラーニングのフレームワークでは Python から操作できるインターフェースが提供されている[62]。本実験では Python を使ってディープラーニングによるシステムを実装する。

3.4.2 データ内容

以下の 2 種類の LIB 正極の SEM 画像を用いる。各画像は三次元の X 軸, Y 軸, Z 軸方向から撮影したものである。熟練者により人手でビットマップ上に空隙箇所をマーキングして 1127 枚の画像の空隙の教師データを作成し、学習用データとして使用する。

【空隙抽出用データ 1】

- リチウムイオン電池 LB1
 - オリジナル SEM 画像
 - X 軸方向画像 : 280 枚 サイズ : 306x200 pixel
 - Y 軸方向画像 : 306 枚 サイズ : 200x280 pixel
 - Z 軸方向画像 : 200 枚 サイズ : 306x280 pixel
 - 空隙の教師画像
 - X 軸方向画像 : 280 枚 サイズ : 306x200 pixel
 - Y 軸方向画像 : 306 枚 サイズ : 200x280 pixel
 - Z 軸方向画像 : 200 枚 サイズ : 306x280 pixel

【空隙抽出用データ 2】

- リチウムイオン電池 LB2

- オリジナル SEM 画像
 - X 軸方向画像 : 341 枚 サイズ : 346x300 pixel
- 空隙の教師画像
 - X 軸方向画像 : 341 枚 サイズ : 346x300 pixel

3.4.3 データ前処理

ディープラーニングを実行する前処理として, LIB 画像の特徴と空隙領域の情報を効果的に抽出するために, 入力画像データを最適なサイズに分割する. 分割されたデータはすべて同じサイズとし, 粒子の表面周辺の特徴を失わずに, 一つの画像に粒子のエッジが入るよう, 粒子面積 $X = 1, \dots, m$ から式(3.7), (3.8)を用いて平均値 U と平方根 W を求め, W 値からデータの分割サイズを決定する.

$$U = \frac{1}{m} \sum_{i=1}^m X_i \quad (3.7)$$

$$W = \sqrt{U} \quad (3.8)$$

本実験で使用する SEM 画像データ図 3-9 (a)から本システム上で自動的に粒子抽出しナンバーリングする. 図 3-9 (b)に抽出された粒子とナンバーリングされた結果を示す. 赤い点は重心を表す. この情報を用いて各粒子面積 X_m が求められる.

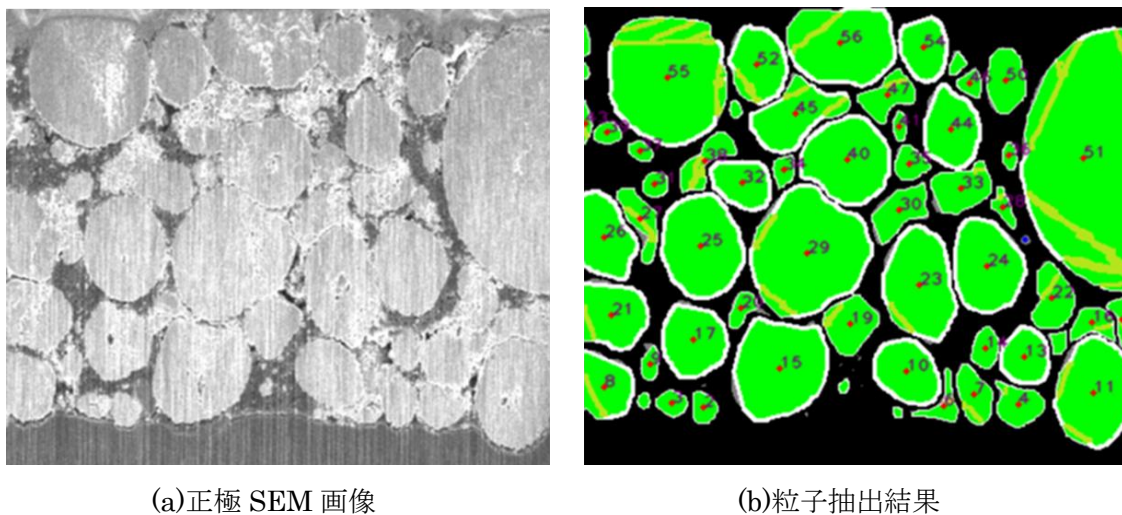


図 3-9 LIB 正極の粒子抽出結果

Fig.3-9 Particle extraction result of electrode reaction in LIB Cathode.

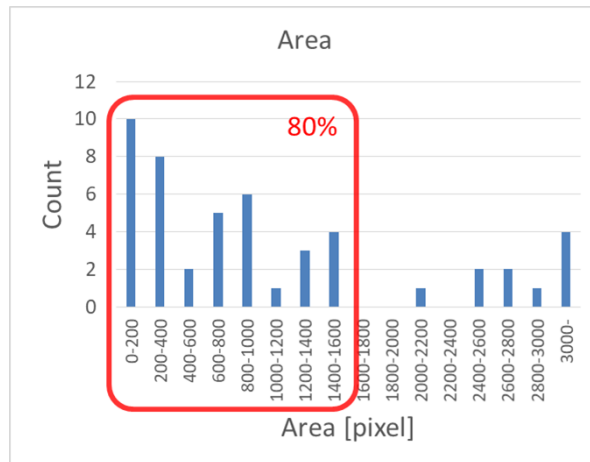


図 3-10 粒子面積分布グラフ

Fig.3-10 Histogram of particle area (pixel).

次に、式(3.7), (3.8)により $U=1300$, $W=36$ (ピクセル)が求まり、 W 値と図 3-10 に示す粒子面積分布グラフを参考に、80%以上の粒子が画像に収まる範囲の 1600 ピクセル(40x40)を分割データサイズとして決定する。データにより、大きな粒子があった場合に、分割したデータに粒子のエッジが入っていない場合がある。例えば図 3-9 (b)の右端にある大きな粒子番号 51 周辺データで空隙情報が入らないケースが考えられる。その場合は教師データから排除して、学習効率を高める。したがって空隙の教師データ数 1127 個を分割することで、各ネットワークモデルが学習に使用したトレーニングデータ数は 28365 個になる。図 3-11 に分割された入力画像データの例を示す。

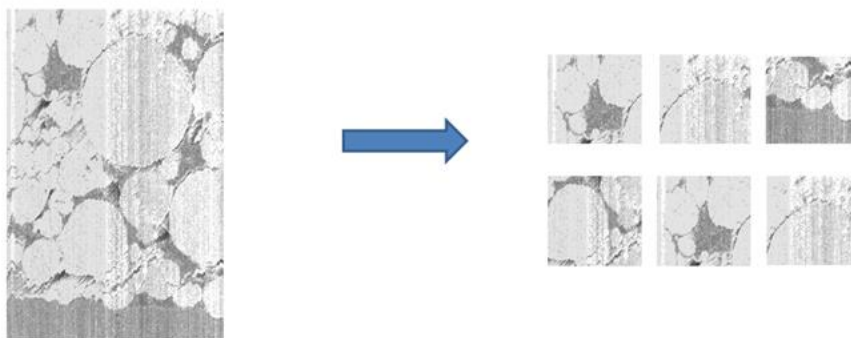


図 3-11 入力画像データ例

Fig.3-11 An image of the input data.

3.5 本手法の検証結果と考察

本節では、実験結果の評価方法は TPR, FPR, AUC, Pixel Accuracy, および IoU の 5 つの定量的指標を用いて抽出精度を比較する。

3.5.1 実験結果

(1) FCN モデル 1

FCN モデル 1 のネットワークモデル(FCN1)について、下記の使用条件でパラメータを設定し計算機実験を行った。

【使用条件】

- 画像データ数：760
- 画像パッチ数：29365
- トレーニングデータ数：28365
- テストデータ数：1000
- エポック数：40
- バッチサイズ：100
- 所要学習時間：約 352 秒=約 6 分 52 秒

図 3-12 に FCN1 における ROC 曲線と定量評価結果を示す。TPR と FPR, IoU の値は閾値が 50%の時の値である。Pixel Accuracy は 92.24%と高いが IoU は 30.28%と低く、本モデルでは精度が不十分である。図 3-13 に FCN1 における Pixel Accuracy の遷移を示す。Pixel Accuracy の遷移について、15~20 エポック目あたりでテストデータに対する精度の上昇がほぼ収束しており、このまま学習させても効果が見られなかった。

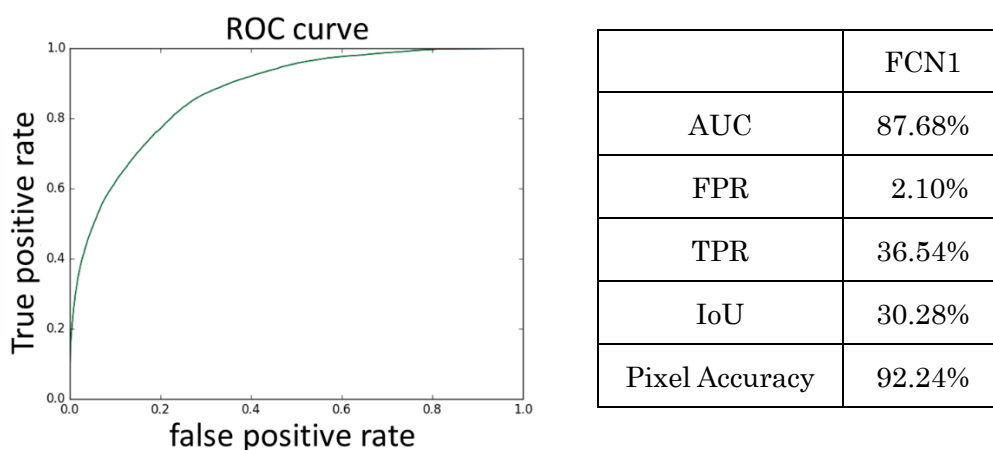


図 3-12 FCN1 における定量評価結果

Fig.3-12 The quantitative evaluation result of FCN1.

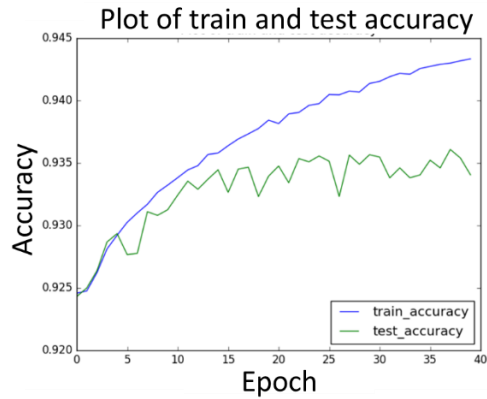


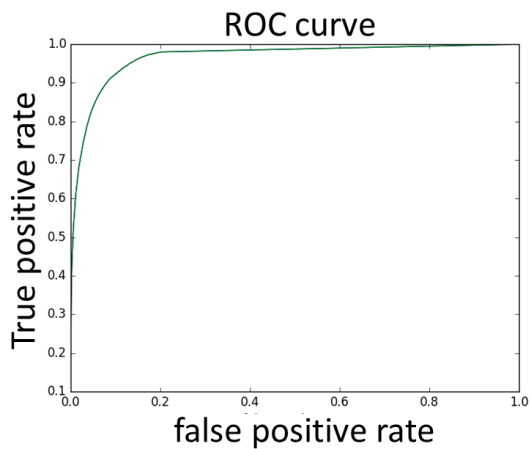
図 3-13 FCN1 における Pixel Accuracy の遷移
Fig.3-13 Pixel Accuracy of FCN1.

(2) FCN モデル 2

FCN モデル 2 のネットワークモデル(FCN2)について、下記の使用条件でパラメータを設定し計算機実験を行った。

【使用条件】

- 画像データ数：760
- 画像パッチ数：29365
- トレーニングデータ数：28365
- テストデータ数：1000
- エポック数：40
- バッチサイズ：100
- 所要学習時間：約 7021 秒=約 1 時間 57 分



	FCN1	FCN2
AUC	87.68%	95.99%
FPR	2.10%	0.86%
TPR	36.54%	59.59%
IoU	30.28%	52.71%
Pixel Accuracy	92.24%	96.68%

図 3-14 FCN2 における定量評価結果
Fig.3-14 The quantitative evaluation result of FCN2.

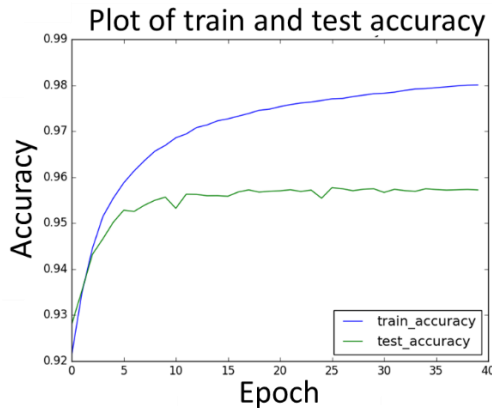


図 3-15 FCN2 における Pixel Accuracy の遷移
Fig.3-15 Pixel Accuracy of FCN2.

図 3-14 に FCN2 における ROC 曲線と定量評価結果を示す。FCN2 は FCN1 と比較し、Pixel Accuracy で約 4%、IoU で約 22%、AUC で約 8%良くなっており大幅に改善された。しかし学習時間は FCN1 の 20 倍を要した。図 3-15 に FCN2 における Pixel Accuracy の遷移を示す。Pixel Accuracy は 10 エポック程度で上昇がほぼ収束しており、このまま学習させても効果が見られなかった。

(3) U-Net モデル

本提案ネットワークモデル(U-Net)について、下記の使用条件でパラメータを設定し計算機実験を行った。

【使用条件】

- 画像データ数：1079
- 画像パッチ数：47285
- トレーニングデータ数：60170（ロバスト対策のデータを含む）
- テストデータ数：1300
- エポック数：40
- バッチサイズ：100
- 所要学習時間：約 15478 秒=約 4 時間 18 分

本手法の独自の機能としてデータ拡張(ロバスト対策)を行い、ディープラーニングのロバスト性を向上させた。これら処理されたデータをトレーニングデータに追加することで、FCN1, FCN2 のトレーニングデータ数 28365 個に対して約 2 倍の 60170 個で学習した。学習するデータ数が多くなった結果、学習時間も FCN2 と比較して 2 倍以上かかった。

図 3-16 に U-Net における ROC 曲線と定量評価結果を示す。U-Net は FCN2 と比較し、すべての指標で改善された。特に IoU で 15%、AUC は 98%超良くなっており十分な精度

を実現できたと考えられる. 図 3-17 に U-Net における Pixel Accuracy の遷移を示す. Pixel Accuracy は 25 エポック程度で上昇がほぼ収束しており, このまま学習させても効果が見られなかった.

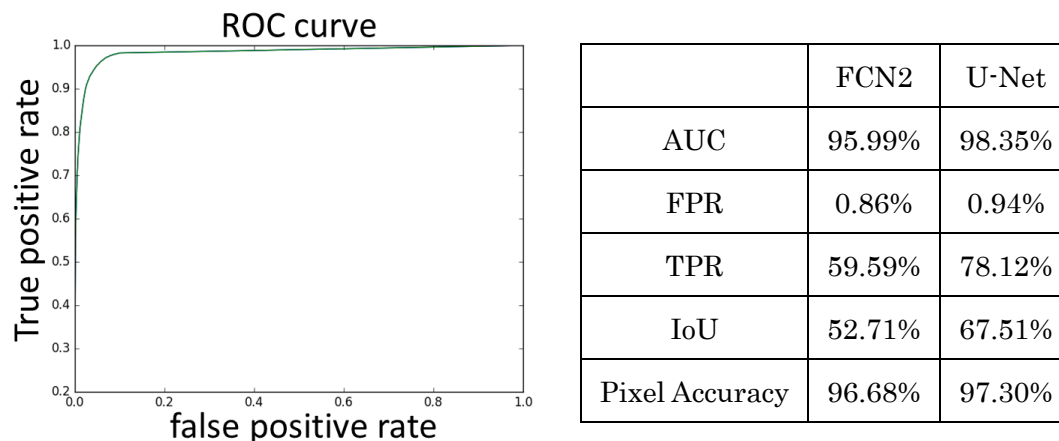


図 3-16 U-Net における定量評価結果

Fig.3-16 The quantitative evaluation result of U-Net.

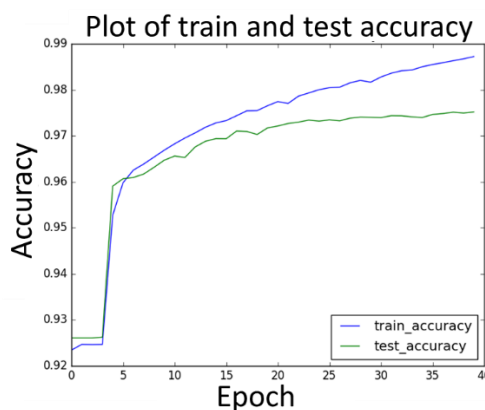


図 3-17 U-Net における Pixel Accuracy の遷移

Fig.3-17 Pixel Accuracy of U-Net.

3.5.2 ネットワークモデルの精度比較

AUC, TPR, IoU, Pixel Accuracy の 4 つの評価指標から, 我々の採用した U-Net の優位性を確認できた. チャンネル数を最大 512 まで増やして畳み込みとプーリングを繰り返しエンコードすることでロバスト性を増し, しかも完全に左右対称な形の Encoder - Decoder 構造を持ち, エンコード時に失われた解像度の特徴マップ情報を補いながらデコードするので, これだけの抽出精度の差異を出すことができた. 表 3-2 に各モデルの空隙抽出精度の結果を示す.

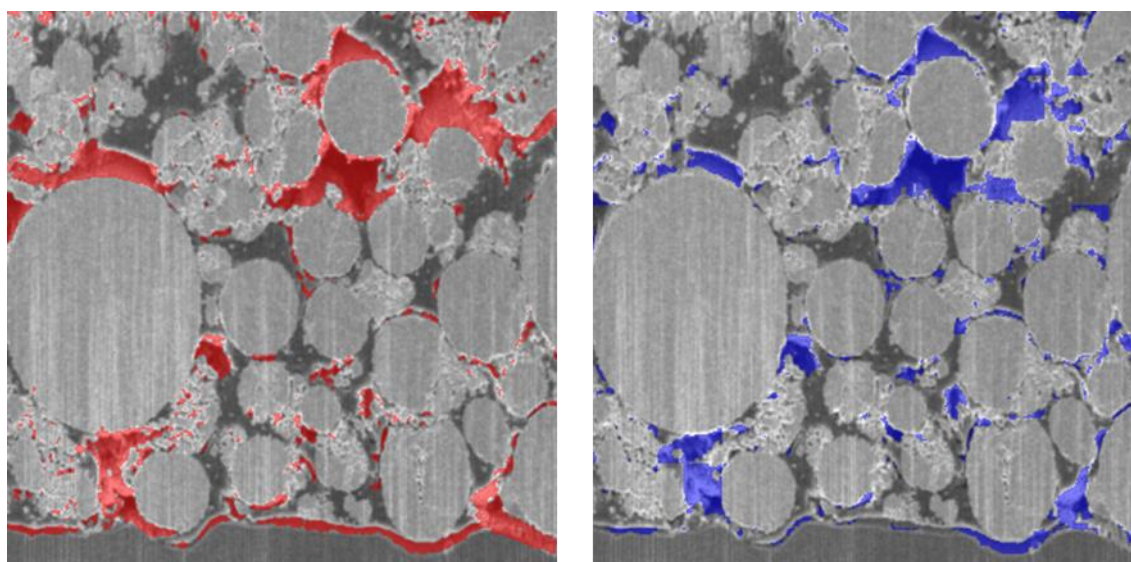
表 3-2 各モデルの精度結果

Table 3-2 The accuracy result of each model.

	FCN1	FCN2	U-Net
AUC	87.68%	95.99%	98.35%
FPR	2.10%	0.86%	0.94%
TPR	36.54%	59.59%	78.12%
IoU	30.28%	52.71%	67.51%
Pixel Accuracy	92.24%	96.68%	97.30%

3.5.3 人による画像抽出

本手法による空隙抽出精度と熟練者による精度を比較するために、別の LIB 画像データを用いて材料開発者の人手による空隙抽出との比較を行った。



(a)正解データ(人手抽出)
(赤：空隙箇所)

(b)出力結果(自動抽出)
(青：空隙箇所)

図 3-18 U-Net による空隙の抽出結果

Fig.3-18 The pores detection results of U-Net model.

被験者は初心者（入社 2 年）から熟練者（入社 25 年）までの材料開発者 10 名で構成される。ニューラルネットワークには同画像を学習させていない U-Net を適用した。図 3-18 に空隙の抽出結果として(a)が正解データ（人が作成）、(b)が本提案手法による自動抽出結果を示す。正解データと比較して、視覚的に見てもほぼ同等の結果が得られていることが分かる。

次に熟練者が作成した空隙抽出データと予測結果のピクセル単位の重なりから IoU で精度比較した。図 3-19 で箱ひげ図により人による抽出精度のばらつきを比較した。

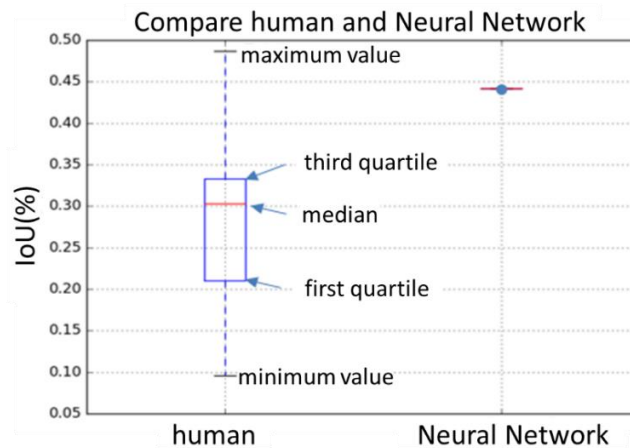


図 3-19 箱ひげ図による人とニューラルネットワークとの空隙抽出性能の比較

Fig.3-19 Comparing human with the neural network for accuracy of pores extraction by Box plot.

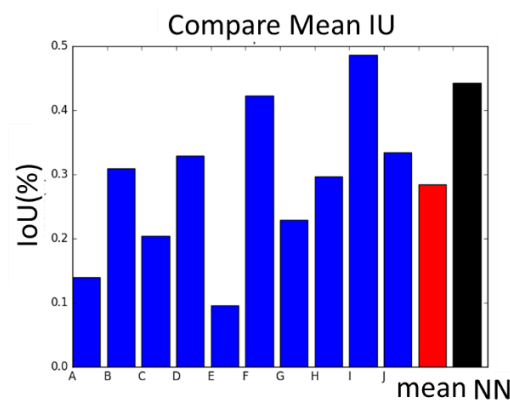


図 3-20 個人別ニューラルネットワークとの空隙抽出性能の比較

Fig.3-20 Comparing individual human with the neural network for accuracy of pores detection.

図 3-20 で棒グラフにより個々の精度を比較した。ここで青色が開発者、赤色が同 10 名の抽出精度の平均値” mean”，黒色が本提案手法” NN” を表す。経験の違いにより被験者間で精度に 2~5 倍の差が確認され、空隙を見分けるには電池材料の立体的構造をイメージする経験的判断が必要であると考えられる。被験者の IoU 精度の平均値(mean) 30.1%に対して本手法は約 1.5 倍の 44.4%で抽出した。熟練者の最も高い精度 49.2%で熟練者のレベルを超えることはできなかったが、他と比較するとほぼ同等の精度が実験的に確認できた。抽出作業は 1 画像当たり人手で 1 時間程度、本手法は数秒程度で完了した(学習時間除く)。

図 3-21 に個人別の人手による空隙抽出結果(黒:空隙箇所)を示す。人によってかなりの空隙認識にバラツキがあることがわかる。このことから空隙を見分けるには電池材料の立体的構造がイメージできる経験に裏付けされた判断が必要であると考えられる。

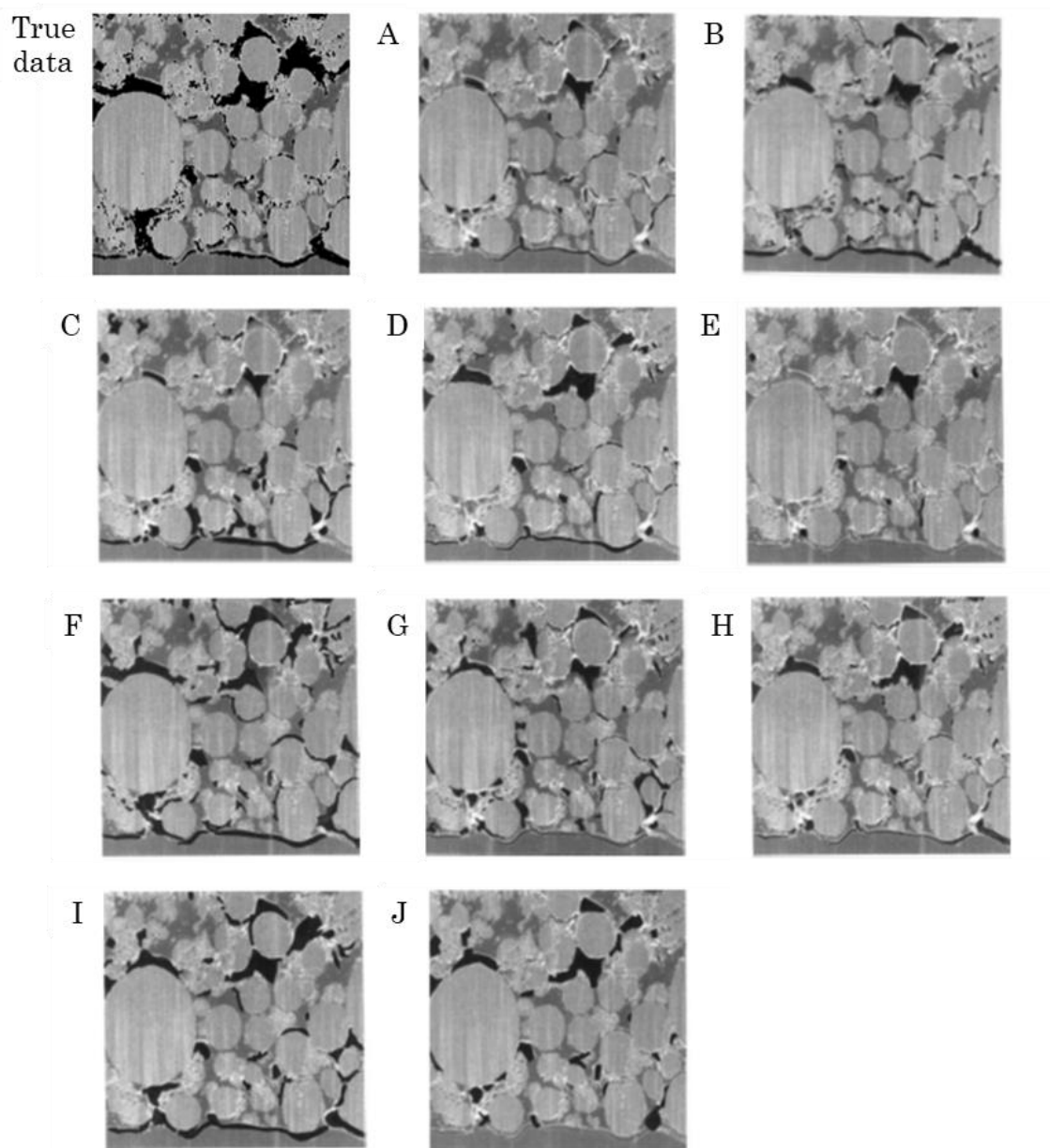


図 3-21 個人別人手による空隙抽出結果の比較（黒：空隙箇所）

Fig.3-21 The pores detected by human (black areas).

3.6 まとめ

本章では、これまでに蓄積した LIB 正極の SEM 画像データから MI で活用するために必要なデータを自動抽出する方法として、ディープラーニングを用いた画像解析による空隙の自動抽出アルゴリズムを提案した。熟練者が作成した空隙抽出データと予測結果のピクセル単位の重なりから IoU (Intersection over Union) 評価指標で精度比較した。被験者は初心者（入社 2 年）から熟練者（入社 25 年）までの材料開発者 10 名で構成する。経験の違いにより被験者間で精度に 2~5 倍の差が確認できたことから、空隙を見分けるには電池材料の立体的構造がイメージできる経験に裏付けされた判断が必要であると考えられる。被

験者の IoU 精度の平均値(mean) 30.1%に対して本手法は約 1.5 倍の 44.4%で抽出した。最も高い精度 49.2%の熟練者のレベルを越すことはできなかったが、かなり近い精度を実験的に確認できた。抽出作業は 1 画像当たり人手で 1 時間程度、本手法は数秒程度で完了した(学習時間除く)。また、本手法の IoU 精度は、標準 FCN モデル 30.3%、FCN 改モデル 52.7%に比べ 67.5%であり、明らかな優位性が見られた。以上のことから本手法による空隙抽出の自動化による作業効率向上への有効性が確認され工学的に有用であると考えられる。

空隙は電池内の空洞になっている部分を指し、電解液が溜まっていた領域を表す。LIB 材料開発において、開発者は SEM 画像から、粒子や活物質の大きさや、活物質などの粒子に付着する導電助剤やバインダーの付着状況などの物体認識に加えて、空隙領域により、粒子の位置関係や奥行きなどの内部構造を把握し、電子やリチウムイオンの通り道を空間認識する。リチウムイオンの通り易さ、内部構成の実現性や、導電ネットワークの最短経路などを制御する内部構成を設計することが可能になる。そのため空隙情報は電池の性能を左右する重要な情報となる。さらには LIB の空隙以外に着目する重要な組成構成や空間状態を数値化することも可能になり、本手法はとても有用性が高いと考えられる。

第4章 深層学習を用いた電池状態推定法

本章では、統計的推定手法を併用して充足した教師データを使い、ニューラルネットワークの中でも時間依存性を利用する RNN の 1 つである LSTM を用いた LIB の SOC 推定方法について述べる。実際の機器使用状態において BMS から出力される端子電圧、電流を用いて統計的推定手法で SOC 推定したものを教師データに充足する。深層学習に必要とされる多様な教師データを大量に準備する方法や、過去の知識モデルを活用することにより学習効率を向上させることで学習時間を短縮する方法を提案する。

4.1 はじめに

電池の SOC 推定方法について、電流積算法とカルマンフィルタ法などの従来技術に対して、最近では深層学習（ディープラーニング）を用いた方法[23]-[25], [58]-[61]が提案されている。ディープラーニングには様々な充放電パターンや周囲温度条件下での SOC 推定に対して、異なるパラメータ設定や回路モデルを必要とせず、少ない周囲温度条件などの学習条件で電池状態の SOC 推定が可能である。しかし電池が組み込まれる実際の運用状況では、充放電パターンや周囲温度など様々な使用環境になり、特に低温時や SOC が低い時の精度への課題[58][61]など、高精度な推定を維持するには、その運用条件に合った適切な教師データを一定数準備する負担が残る。また EKF の推定に対して条件変動に対してディープラーニングと組み合わせてモデルパラメータを設定している方法[59][60]が提案されるが動的条件の場合の精度が不安定で、しかも同様に教師データの準備に課題が残る。また転移学習[55]により学習効率化による精度向上と計算時間削減効果が確認されているが、利用する知識モデルとの関連性に効果が影響されるため転移先との相性など調整事項が多い。本論文では、実際の機器使用状態において、電池管理する BMS で観測される端子電圧や電流などの情報を用いて、統計的推定手法で残量推定したものを教師データに追加する。この教師データを用いてディープラーニングで学習し SOC を推定する方法である。実際の運用状態での多様な電池状態を含む充足された教師データを準備する方法を提案する。さらにこれらの教師データを大量に学習することで教師あり学習の信頼性が向上する。深層学習が持つ高いロバスト性能を活かした高精度な SOC 推定方法を提案する。

4.2 深層学習を活用した高精度電池 SOC 推定

本節では、ディープラーニングを活用した本提案手法を説明する。深層学習の活用には、適切なネットワークモデルの構築、教師データの準備と、学習時間の短縮が課題としてあげられる。これらの課題について順に示す。

4.2.1 ネットワークモデルの構築

本論文で提案するネットワークモデルの構築方法について説明する。一般に蓄電池の抵抗は、温度、SOC、電流などに依存する。BMS などから周期的に測定される端子電圧、電流値、温度はサンプリング時間ごとの時系列データである。時系列データを扱うディープラー

ニングの1つである RNN の中の LSTM を用いて実測追従性の高い回帰モデルを構築することができる。RNN は1ステップ前の隠れ層の状態をフィードバックさせることで、過去の情報を現時点の予測に活かすことができるニューラルネットワークである。図 4-1 に LSTM の展開図を示す。時刻 t でセルは x_t を入力し y_t を出力する。出力 y_t と隠れ状態 h_t は次の時刻 $t+1$ で使用するためにフィードバックさせる。

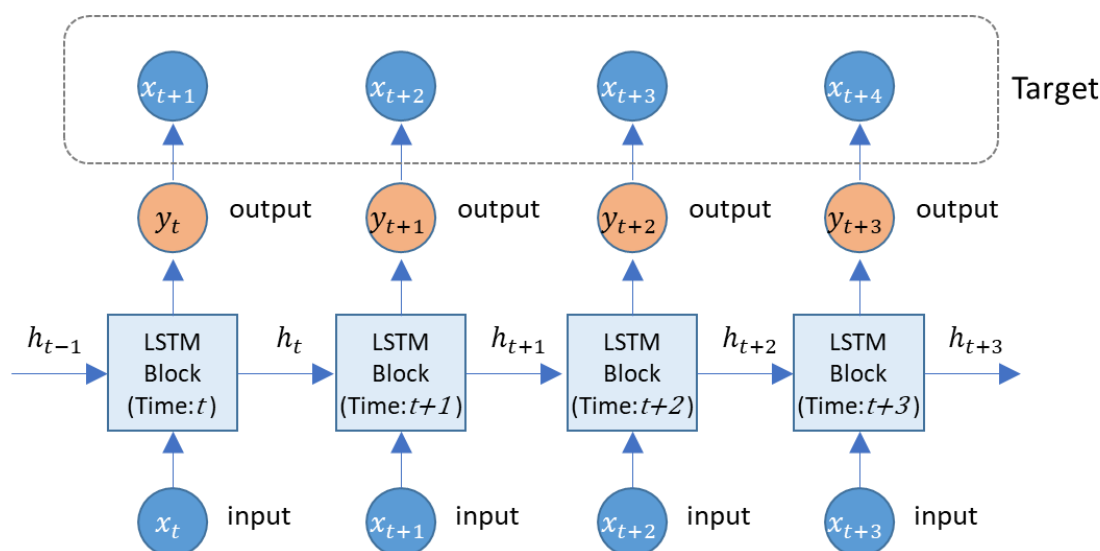


図 4-1 LSTM の展開図

Fig.4-1 Unroll of LSTM.

本論文では、LIB の SOC 推定に用いる LSTM モデルは、入力層、LSTM 層、出力層の3層の基本構成とし、データ入力後、LSTM 層を経て全結合層により線形関数で SOC 推定値を出力する回帰モデルとする。モデルの構築では LSTM 層にディープラーニング向けライブラリ Keras の `Keras.layers.LSTM` を用いる。条件は引数として活性化関数を `'tanh'`、LSTM における各ゲートの活性化関数を `'sigmoid'` に設定した。出力層では全結合層 `Dense` クラスを用いて、条件は活性化関数を `'linear'` に設定した。モデルの学習では誤差関数を `'mean_squared_error'`、学習率を設定する最適化手法に `Adam` を使用して学習率を 0.001 に設定した。これら以外の条件には、一般的に用いられているデフォルト値を用いた[32]。

4.2.2 教師データの準備

実際の電池の充放電パターンや周辺温度環境など様々な使用状況によって劣化の進み具合が異なり、時間に伴って電池性能に関するパラメータも変化する。しかも電池の個体差や状態、とりわけ使用環境に左右される不良状態のデータなど、多様な電池状態における教師データを大量に準備することは難しい。ディープラーニングが多様な環境条件に対してロバスト性に優れた推定方法であっても限界がある。

提案する教師データの構築方法について説明する。高次の非線形モデルを用いてその多くのパラメータを既知として SOC 推定する方法には、昔から広く用いられる EKF による

SOC 推定が提案されている[17]. パラメータが時間とともに変化しない場合に最も効果を発揮するが, 蓄電池の EV や定置用電源として使用される環境では使用条件が時々刻々と変化し, それによりパラメータも変化してしまうため, 状況に合わせてパラメータの調整が必要になる. 一方, 高次の非線形モデルを用いてその多くのパラメータを逐次推定して SOC 推定する方法には RLS を用いた回路パラメータ推定法と EKF を用いたパラメータ依存状態空間モデルを組み合わせた SOC 同時推定法(以下 RLS+EKF 法と表記)による SOC 推定[18][19]が提案されている. 劣化や温度変動に追従した電池モデル変動に対して, RLS によりパラメータを推定して EKF のパラメータ値として入力し, これを用いて EKF で SOC 推定することで, 内部抵抗や OCV の時間的変化に追従した SOC 推定を行うことができる. SOC 同時推定法により, 高次の非線形モデルのパラメータを逐次推定し高精度に SOC 推定することができる. しかし EKF 使用中にノイズ情報の最適値が変動する可能性があり, さらなる精度の安定性が必要とされており, この変動を見極めながら, 精度の高いデータのみ活用することが望ましく, 今後の改良の余地がある.

4.2.3 知識モデルの活用

電池の SOC 推定に過去の知識モデル(ネットワーク構造と重み情報)を活用することで推定精度や計算時間などの学習効率を向上させるよう, 転移学習の利用方法を検討した. 転移学習に使用する学習済み知識モデルが, SOC 推定で対象とする電池と同じもので学習したモデルか, あるいは異なる電池で学習したモデルか, モデルの違いによってどのような学習効果が見られるのかを知る. さらに教師データが少ない場合に, 転移学習によってどのような学習効果が見られるかを知る必要がある. 学習モデルの違いや学習データ量の違いによる転移学習による学習効果を推定精度や計算時間で確認する. さらに転移学習とファインチューニング手法では再学習させる層が異なるため, 計算量が変わる. これらの学習時間と推定精度への影響を確認し, 学習効率を向上させる転移学習活用方法を検討する.

本論文では, 転移学習の学習方法として, 過去に学習した知識モデル(学習済みネットワーク構造と重み)を固定したまま, この後ろのレイヤに層を追加して新たな層の重みのみを学習する転移学習(Transfer Learning, 以下 TL と表記)と, 知識モデルを初期値として, モデル全体の重みを再学習させるファインチューニング(Fine Tuning, 以下 FT と表記)を評価する. 転移学習(TL)とファインチューニング(FT)の学習範囲の違いを図 4-2 に示す.

まず電池の違いによる学習効果を確認するために 2 種類の電池を準備した. 学習用データ (V, I, SOC) を入力して LSTM モデルで学習した知識モデルを作成し, 同モデルを用いて評価用データ (V, I) を入力し SOC 推定値 (\widehat{SOC}) を出力する. 転移学習では過去に学習した知識モデル(重み)を LSTM 層および全結合層の初期値に設定し, LSTM 層の重みは変化させず, 全結合層の重みのみを再学習させることで学習効率を向上させる. 過去に同じネットワーク構造で学習した知識(重み)モデル M を転移学習することで, 推定精度の向上と損失関数の収束時間短縮を図る. この RNN(LSTM)処理では SOC 推定値 $(\widehat{SOC}_{LSTM+est})$ および再学習された知識モデル M' を出力する. 知識モデル M は学習を繰り返すたびに最新の知識

モデル M' に更新される. 入手した新たな運用情報から生成した 2 次データを教師データに追加し再学習することを繰り返すことで高効率高信頼性を可能とする方法を提案する.

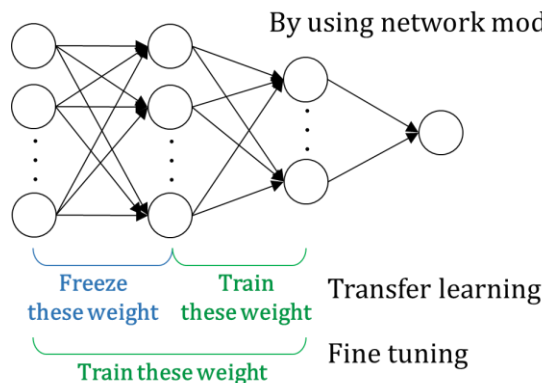


図 4-2 転移学習とファインチューニングの学習範囲
Fig.4-2 Training range of transfer learning and fine tuning.

4.2.4 提案手法

運用状態から出力される測定データを用いているので, 実際に起こりうる多様な電池状態を表す教師データを大量に学習することが可能になる. このことから適切な教師データの準備作業の効率化と現状を反映した充実した内容のデータを確保することが可能になり, よりロバスト性や信頼性の高い SOC 推定が可能になる.

本論文では, まず初めに実験室で得られた LIB 放電試験により測定した端子電圧, 電流, SOC を教師データ Training data_I (V, I, SOC) に用いて RSS(LSTM) により学習する. 過去に学習した知識モデル M があれば, これを転移学習する. 学習後に作成される知識モデル (ネットワーク構造, 重み) M' へ知識モデルデータ M を更新する. 図 4-3 に学習フローを示す. ただし初回は追加する教師データ Training data_II ($V', I', \widehat{SOC}_{RK}$) はまだ準備されていない. 次に提案する教師データの準備方法を説明する. EV などの実際の機器使用状態において BMS から出力される測定可能な端子電圧や電流などの運用情報 (V', I') (1 次データ) を元に LIB 等価回路モデルを介して統計的推定手法 (RLS+EKF 法) により SOC 推定値 (\widehat{SOC}_{RK}) を求め, 運用情報と合わせたデータ Training data_II ($V', I', \widehat{SOC}_{RK}$) (2 次データ) を教師データに追加する. ただし RLS+ EKF 法は精度の安定性に課題があるため, EKF の Algorithm I 中の式(2-15)の事前誤差共分散値 P^- を観測して同分散値の小さい閾値 D 以下の SOC 推定値のみを抽出し教師データに追加することで精度への影響が小さくなるように対応する. 図 4-4 に教師データ構築フローを示す. このようにして 2 次データで構築された多様な電池状態を持つ教師データで学習済みの RNN(LSTM) 予測モデルにより $\widehat{SOC}_{LSTM+est}$ を推定する. 図 4-5 に電池の SOC 推定フローを示す.

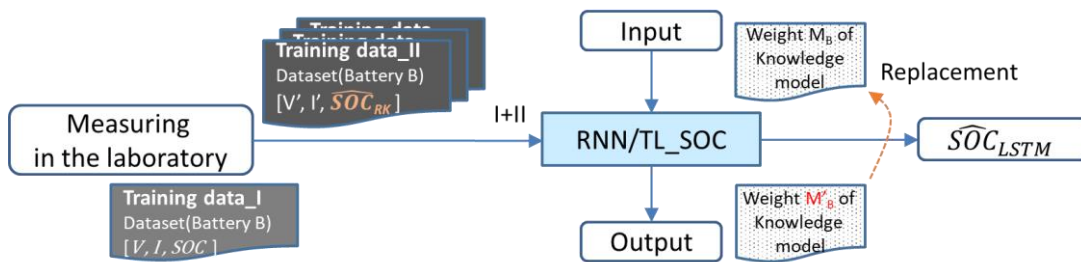


図 4-3 学習の処理フロー

Fig.4-3 Data flow diagram of learning mode.

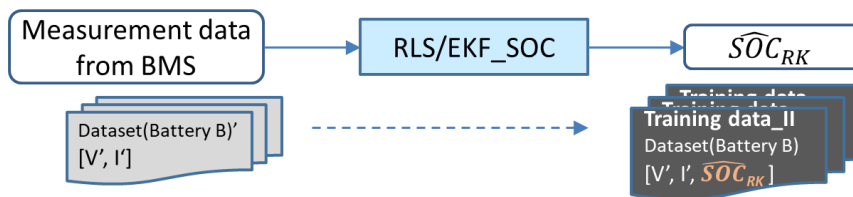


図 4-4 RLS/EKF を用いた教師データ構築方法の処理フロー

Fig.4-4 Data flow of training data construction method using RLS/EKF method.

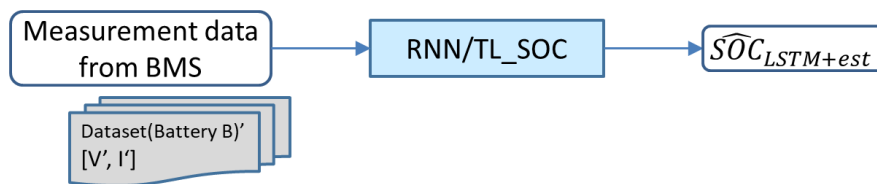


図 4-5 学習済みの RNN(LSTM)予測モデルを用いた SOC 推定方法の処理フロー

Fig.4-5 Data flow of SOC estimation model using learned RNN(LSTM) machine.

本提案手法のアルゴリズムの処理の流れを次の Algorithm II に示す.

Algorithm II

- Step1: 入手した運用情報(V', I')を RLS+EKF 処理へ入力
- Step2: RLS+EKF 処理で SOC 推定値(\widehat{SOC}_{RK})を出力
- Step3: P^- が閾値 D 以下の推定値($V', I', \widehat{SOC}_{RK}$)を教師データへ追加し, RNN(LSTM) 処理へ入力
- Step4: 転移学習に知識モデル M を RNN(LSTM)処理へ入力
- Step5: RNN(LSTM)処理で SOC 推定値($\widehat{SOC}_{LSTM+est}$)と学習した知識モデル M' を出力
- Step6: 知識モデル $M = M'$ に更新し Step1~Step5 を繰り返す

多様な電池の使用状態の教師データを大量に準備し, ディープラーニングに用いることでロバスト性の高い安定した精度を確保する. 従来でも転移学習の効果は報告されている

が、稼働状態の情報から順次教師データが補充され、電池劣化状態を含む電池状態の変化を再学習することでモデルが更新され、さらなる知識モデルの充実と学習効率向上による精度向上、計算時間短縮を図ることができる考える。

電池の SOC 推定に過去の知識モデルを活用することで学習効率を向上させるよう、転移学習の利用方法を検討した。転移学習に使用する知識モデルが、SOC 推定で対象とする電池と同じもので学習したモデルか、異なる電池で学習したモデルかによる計算時間や精度への影響を確認する。また転移学習は手持ちのデータセットが少ない場合に特に有効な手法として用いられる。教師データの学習データ量の違いによる推定精度への影響と、学習データ量が違っても知識モデルを活用した転移学習による学習効果を推定精度で確認する。さらに転移学習とファインチューニング手法では再学習させる層が異なるため、計算量が変わる。これらの学習時間の違いと推定精度への影響を確認し、学習効率を向上させることを検討する。

学習用データ(V, I, SOC)を入力して LSTM モデルで学習した知識モデルを作成し、同モデルを用いて評価用データ(V, I)を入力し SOC 推定値(\widehat{SOC})を出力する。転移学習では過去に学習した知識モデル(重み)を LSTM 層および全結合層の初期値に設定し、LSTM 層の重みは変化させず、全結合層の重みのみを再学習させることで学習効率を向上させる。

4.3 評価指標

LSTM の有効性を評価するために、次の評価指標を用いる。本提案方法によって得られた SOC 推定値の精度を検証する。SOC 実測値を真値とし SOC 推定値との誤差を検証する。各手法による SOC 推定精度の比較には、評価指標として要素数を n 、 i 番目の実測値を y_i 、推定値を \hat{y}_i とする式(4.1)の平均平方二乗誤差 RMSE(Root Mean Square Error, 以下 RMSE と表記)を用いる。

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (4.1)$$

4.4 本手法の実験方法

4.4.1 開発環境

本実験に使用した開発環境のハードウェア、ソフトウェア構成は以下の通りである。

【ハードウェア】

- CPU : Intel®core™ i7-8700CPU@3.20GHz
- RAM : 16GB

【ソフトウェア】

- OS : Windows10 Enterprise

- プログラミング言語 : Python 3.5
- ディープラーニングライブラリ : TensorFlow 2.3.0, Keras 2.4.3

本実験ではプログラム言語に Python を使用する。Python は科学分野、特に機械学習やデータサイエンスの分野でよく使われており、NumPy や SciPy といった数値計算や統計処理を行うライブラリや、Matplotlib といったグラフ描画のためのライブラリや、Google 製の TensorFlow などのディープラーニングのフレームワークでは Python から操作できるインターフェースが提供されている。Keras は TensorFlow などをバックエンドにして動作するライブラリである。本実験では、Python を使ってディープラーニングによる本提案手法を実装する。

4.4.2 データ内容

提案手法の有効性を確認するために、計算機実験を行った。実験に使用した電池は市販で入手しやすい Ni・Co・Mn の三元系を正極に用いた 2 種類の円筒型 Panasonic 製 18650 型 LIB[63][64]である。同 LIB 仕様を表 4-1 に示す。これら 2 種類の LIB (LIB A, LIB B)について LIB 放電試験により測定した端子電圧、電流、SOC を教師データとして使用する。

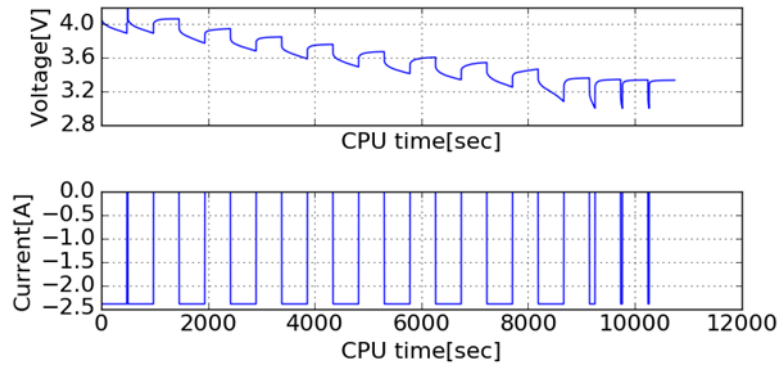
表 4-1 実験で使用した 18650 型 LIB のスペック

Table 4-1 18650 type LIBs specifications used in this experiment.

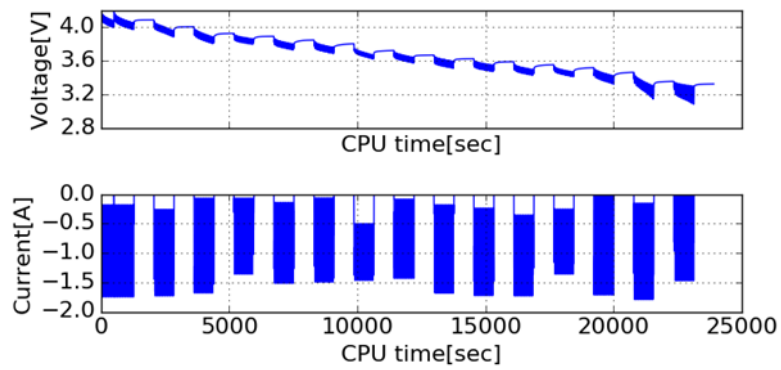
	LIB A (UR18650AA) [63]	LIB B (NCR18650BF) [64]
公称容量	2250[mAh]	3350[mAh]
公称電圧	3.6[V]	3.6[V]
カットオフ電圧	3.0[V]	3.0[V]
最大電圧	4.2[V]	4.2[V]

放電レートは LIB A では 1 種類(1C)、LIB B では 3 種類(1C, 0.5C, 0.75C) について、放電パターンは模擬的にパルス放電、ランダム波形放電、三角波形放電の 3 種類の放電試験を 25°C の温度環境において実施し、端子電圧、電流、SOC を測定した。これらの LIB の基本的な特性データを測定して教師データを作成し、基本的な特性の特徴量を学習することで、実際に LIB が使用される様々な放電パターンがこれらの組み合わせ問題として対応可能になる。さらに本提案手法で準備する 2 次データを教師データに追加することで、様々な温度条件による特性変化や不良条件による特性などを教師データとして再学習することができ、対応可能になると考える。図 4-6 に LIB B の各放電パターンの放電試験における端子電圧と電流の一例を示す。

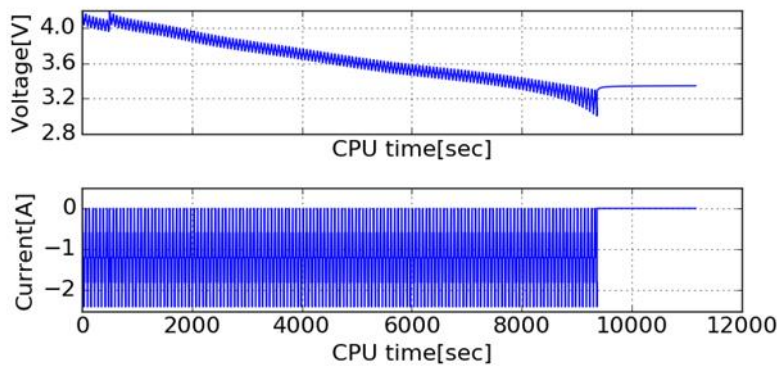
LIB A にはパルス放電データ 2 個、ランダム波形放電データ 1 個、三角波形放電データ 1 個の計 4 個のデータを準備した。LIB B には 3 種類の放電レートに対して、それぞれ以下のようにデータを準備した。



(a) Pulse discharger.



(b) Random discharger.



(c) Triangular-wave discharger.

図 4-6 各放電試験における端子電圧と電流
Fig.4-6 The battery's terminal voltage and current.

- パルス放電では 480 秒周期に一定の電流を放電したデータと 360 秒周期に一定の電流を放電したデータと 240 秒周期に一定の電流を放電したデータの (3 種類の周期) × (3 種類の放電レート) = 9 個のデータを準備した.
- ランダム波形放電では 8 秒毎にランダムに出力電流を変えて 10 分周期で放電と停止を繰り返したデータと, 6 秒毎にランダムに出力電流を変えて 10 分周期で放電と停止を繰り返したデータと, 4 秒ごとにランダムに出力電流を変えて 10 分周期で放電と停止を繰り返したデータの (3 種類の周期) × (3 種類の放電レート) = 9 個のデータを準備した.
- 三角波形放電では電流の波形が三角になるよう 8 秒毎に出力電流を調整して 10 分周期で放電と停止を繰り返したデータと, 6 秒毎に出力電流を調整して 10 分周期で放電と停止を繰り返したデータと, 4 秒毎に出力電流を調整して 10 分周期で放電と停止を繰り返したデータの (3 種類の周期) × (3 種類の放電レート) = 9 個のデータを準備した.

これら合計(3種類の放電パターン) × (9個のデータ) = 27個のデータを準備した.

本システム実装上, 学習用データに対して前処理としてデータを平均値 0, 標準偏差 1 のデータ [0, 1] の範囲に標準化する. 標準化データ X はデータ: x , 平均値: μ , 標準偏差: σ として, 以下の式(4.2)で計算される.

$$X = (x - \mu) / \sigma \quad (4.2)$$

RNN モデルを作成するにあたり, データ準備として RNN をタイムステップで展開する. 入力データをタイムステップ T で分割し 1 段シフトして再度配列しなおして, 行列 (T , データ長) の学習用データセットを生成することで, タイムステップ時間を同時処理しサンプリング時間の推移でシフトさせ同時処理する.

4.4.3 予備実験

LSTM モデルの使用には各種ハイパーパラメータ条件の設定が必要である. ハイパーパラメータはデータ依存の要素が大きく, パラメータチューニングも重要な課題の 1 つであるが, ここでは本手法の学習効果を確認するため議論の本質とせず, パラメータ条件を仮決定し固定して同条件で評価を行う. まずパラメータの中でも様々な分野で一般的に用いられている値のバッチサイズ 8, サンプル周期 1 秒, 学習率 0.001 に設定する[62]. 次に対象とするデータの特徴によって影響を受けやすい次の 3 つ, エポック数, 隠れ層数, タイムステップについて, 予備実験にて順に仮固定して推定精度を評価し, 推定性能への影響範囲を確認して, 良好な精度と処理時間が確保できるパラメータ条件の組み合わせを仮決定する.

4.5 本手法の検証結果と考察

本節では, ネットワークモデルにおいて SOC 推定精度への影響が大きいハイパーパラメータ (エポック数, 隠れ層数, タイムステップ) の条件を決定するための評価実験を行う.

同パラメータの中で条件によって影響の受けやすい変動の大きいものを事前に評価実験して推定精度への影響範囲を確認した。良好な精度と処理時間が確保できるパラメータ条件を決定し固定した上で、次に本手法による教師データ構築方法や SOC 推定方法と知識モデルを活用して学習効率を上げる方法について提案する。

4.5.1 予備実験結果

LSTM モデルの学習に使用されるハイパーパラメータ条件を仮決定するために、LIB B のデータを用いてバッチサイズ 8、サンプル周期 1 秒、学習率 0.001 に設定[62]し比較実験した。同パラメータの中でも条件により変動の大きいエポック数、隠れ層数、タイムステップについて推定精度への影響範囲を確認した。エポック数は対象データ全体を繰り返し学習する回数であり、隠れ層数は LSTM モデルの隠れ状態の深さを表す。タイムステップは異なる入力に対して同時に時間依存性を見る時間範囲である。学習におけるエポック数毎の損失関数の収束状況で学習の進み具合を確認した。どの放電パターンにおいてもエポック数 15 には概ね収束することが確認できたためエポック数を 15 に決定した。

次にモデルの隠れ層数とタイムステップと推定精度の関係性を確認する。隠れ層数 (HID_LY) を 3, 30, 50, 100, 150, タイムステップ(maxlen) を 5, 50, 500, 1000, 2000 で評価した。評価用データにパルス放電データを用いた時の精度結果をそれぞれ表 4-2, 4-3 に示す。

表 4-2 隠れ層数(HID_LY)別の SOC 推定値(%)精度

Table 4-2 Accuracy of SOC estimates for hidden layers at maxlen=500.

HID_LY	3	30	50	100	150
RMSE	2.2	1.2	1.7	2.7	3.3
CPU time[s]	365	1213	1983	4042	7982

表 4-3 タイムステップ(maxlen)別の SOC 推定値(%)精度

Table 4-3 Accuracy of SOC estimates for time steps at HID_LY=30.

maxlen	5	50	500	1000	2000
RMSE	4.0	1.4	1.2	1.7	3.0
CPU time[s]	236	401	1213	1960	3752

隠れ層数やタイムステップを増やすことが必ずしも学習効果につながらず、逆に精度が悪化し処理時間は指数関数的に増加していた。隠れ層数ではネットワークの複雑さや過学習があり、またタイムステップでは大きければ過去の情報も同時に多くの特徴量で考慮できるが近傍の情報が重要であり弊害も生まれる。

どの放電パターンでも隠れ層数 30, データ分割長 50~1000 で誤差分散が安定し, 概ね誤差 2%以内を確認できたため, 隠れ層数を 30, データ分割長 500 に決定した. CPU time は LSTM の学習時間である.

4.5.2 実験結果

本実験では, 観測可能な 1 次データ(V, I)をもとに RLS+EKF 法で得られた SOC 推定値 (2 次データ) を学習用データに追加する学習用データ準備方法の有効性と, 転移学習による学習効果の影響を確認した.

(1) 学習用データ数による学習効果

学習用データ数による学習効果の違いを確認する. LIB B から 3 種類の放電パターン毎に, ランダムに 1 個データを選択し $3 \times 1 = 3$ 個を評価用データに設定し, 残り 8 個のデータ $3 \times 8 = 24$ 個を学習用データに用いて LSTM 学習による SOC 推定(LSTM24)を行う. またこの 24 個のデータから 3 種類の放電パターン毎にランダムに 4 個のデータを選択し $3 \times 4 = 12$ 個のデータを学習用データに用いて, LSTM 学習による SOC 推定(LSTM12)を行い, 学習用データ数の違いによる誤差比較を図 4-7 に示す. どの放電パターンでも学習用データ数が多いほうが精度の高いことを確認できた. また学習時間はデータ数に線形比例して 2 倍かかった.

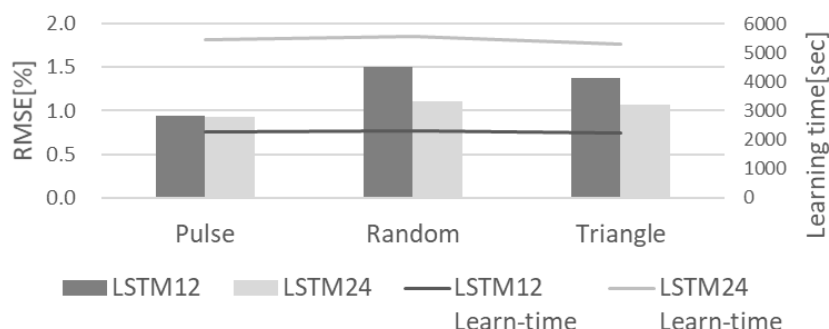


図 4-7 学習用データ数の違いによる誤差比較

Fig.4-7 Comparison of estimations by the difference in the number of training data (12 or 24).

(2) 統計的推定手法による推定値の活用方法

統計的推定手法である RLS+EKF 法による SOC 推定精度を確認した. LIB B から 3 種類の放電パターン毎にランダムに 1 個のデータを選択し $3 \times 1 = 3$ 個を評価用データに設定し, 残り 8 個のデータ $3 \times 8 = 24$ 個を学習用データに用いて, RLS+EKF 法による推定 (RLS+EKF24) と LSTM (LSTM24) による推定精度を比較した. 図 4-8 に結果を示す.

実験結果では RLS+EKF 法だと放電パターンによって RMSE が 0.7%~2.0%と精度にバラツキが見られた. 本手法ではすべての放電パターンで RMSE が 1.0%の安定した良好な

精度を確認でき、どちらの方法でも推定時間は同じような結果になった。但し、LSTM ではこれとは別に多くの学習時間が必要になる。

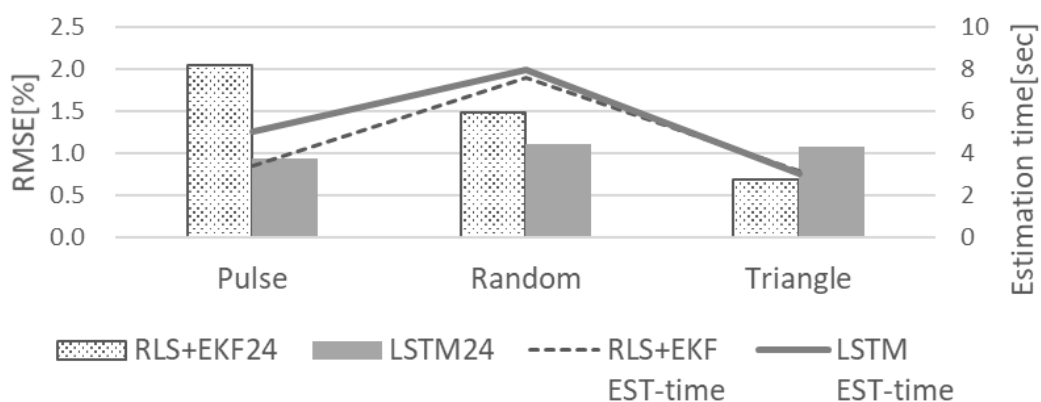


図 4-8 RLS+EKF 法と LSTM による推定精度比較

Fig.4-8 Comparison of estimations by RLS+EKF and LSTM.

観測可能な 1 次データ (V, I) をもとに RLS+EKF 法で得られた SOC 推定値 (2 次データ) を教師データに活用するために SOC 推定精度のバラツキ具合を確認した。この 24 個のデータについて RLS+EKF 法による SOC 推定した結果を図 4-9 に示す。RLS+EKF 法では RMSE が 0.5~3.0% と精度にバラツキが見られる。

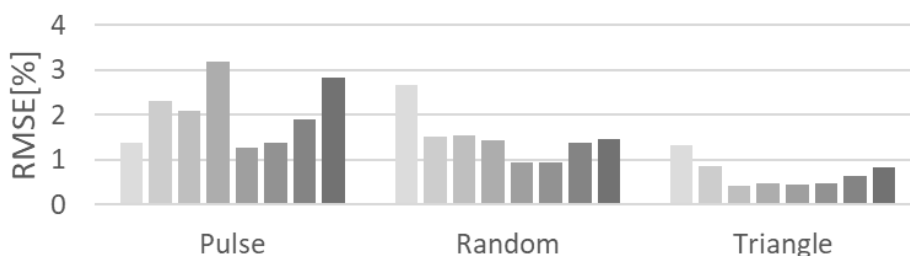
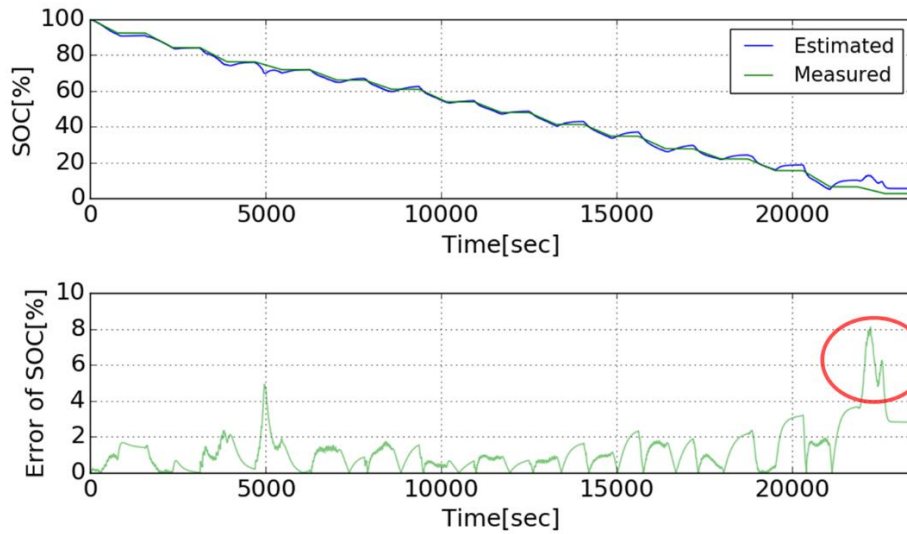


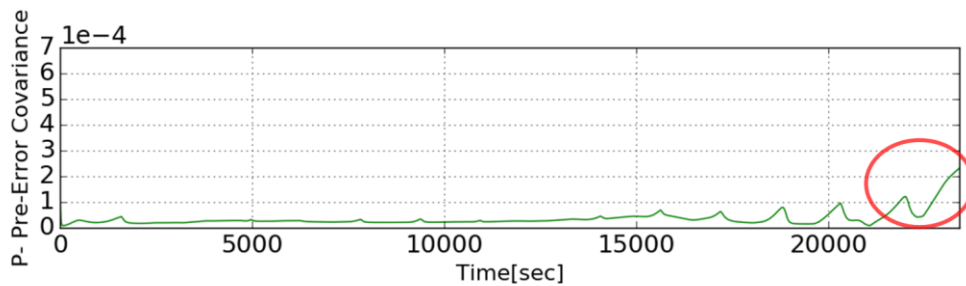
図 4-9 RLS+EKF 法による SOC 推定精度

Fig.4-9 Accuracy of SOC estimations by RLS+EKF.

SOC 推定精度の高い部分のみを教師データに活用する。推定精度の判定には、SOC 推定値の誤差変動に対して、第 2 章 カルマンフィルタの Algorithm I の予測ステップ中の式 (2.15) の事前誤差共分散値 P^- をモニタリングし、閾値 ($D = 1e - 4$) を設けてこれより分散値の小さい時の SOC 推定値のみを抽出し教師データに追加した。図 4-10 にランダム波形放電における SOC 推定値の誤差比較を示す。(a) の上図が SOC 実測値と推定値の比較、下図がその絶対値誤差推移、(b) は事前誤差共分散の推移グラフである。赤丸箇所のように同分散値の大きなところを除く時間での推定値を教師データとして活用した。



(a) Error of SOC estimation for random wave.



(b) Estimated error covariance for random wave.

図 4-10 RLS+EKF 法による推定精度と分散

Fig.4-10 Estimation accuracy and error covariance by RLS+EKF.

LIB B から 3 種類の放電パターン毎にランダムに 1 個のデータを選択し $3 \times 1 = 3$ 個を評価用データに設定し、残り 8 個のデータ $3 \times 8 = 24$ 個を学習用データに用いて、LSTM 学習による SOC 推定(LSTM24)を行う。次にその半分の 12 個のデータを RLS+EKF で得た SOC 推定値に置き換えたデータを用いて LSTM 学習による SOC 推定(LSTM+est)を行う。さらに LSTM+est と同じデータを用いて LIB B の別の 12 個のデータで作成した知識モデル M_B を転移学習して、LSTM 学習による SOC 推定(LSTM+ets+TLb)を行う。これら学習データの違いと転移学習の効果を比較し影響を確認した。図 4-11 に結果を示す。

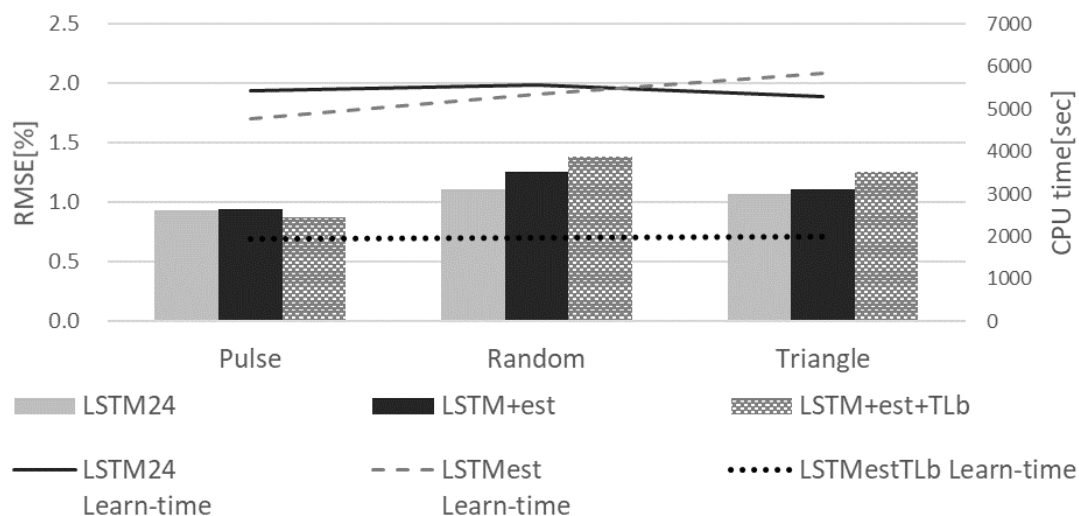


図 4-11 学習データ生成法と転移学習に基づく LSTM による推定精度への影響
 Fig.4-11 Impact on estimations accuracy by LSTM using generated training data with TL.

LSTM24 と LSTM+est を比較すると、RMSE 0.1%程度の精度劣化で抑えられた。RLS+EKF 法の精度の安定性の課題に対して、同法で得られた推定値を追加する際の事前誤差共分散の閾値設定による安定した精度のみ教師データに追加したことと、ロバスト性の高い LSTM による学習効果で、推定精度への影響を最小限に抑え、安定した精度を維持できることが確認できた。

LSTM+est に対して LIB B の学習済み知識モデル M_B を用いて転移学習した LSTM+est+TLb と比較すると、ランダム放電データでは 0.3%程度の精度劣化が見られたが、他の放電パターンでは 0~0.2%内の精度劣化に抑えることができ、全体として概ね良好な結果が得られた。特に学習に要した時間が 50%以下に削減され、転移学習により学習効率が向上した効果を確認できた。

パルス放電パターンの評価データを用いて、学習データの違いや転移学習した場合の SOC 推定値と実測値との誤差グラフを図 4-12, 4-13, 4-14 に示す。上図が SOC 推定値と実測値のグラフ、下図が絶対値誤差推移グラフである。図 4-12 と図 4-13 を比較すると、推定値を学習データに利用しても明らかな精度劣化は見られなかった。また図 4-13 と図 4-14 を比較すると、転移学習により知識モデルの初期値が与えられるため、放電の初期段階から精度よく推定できているのがわかる。

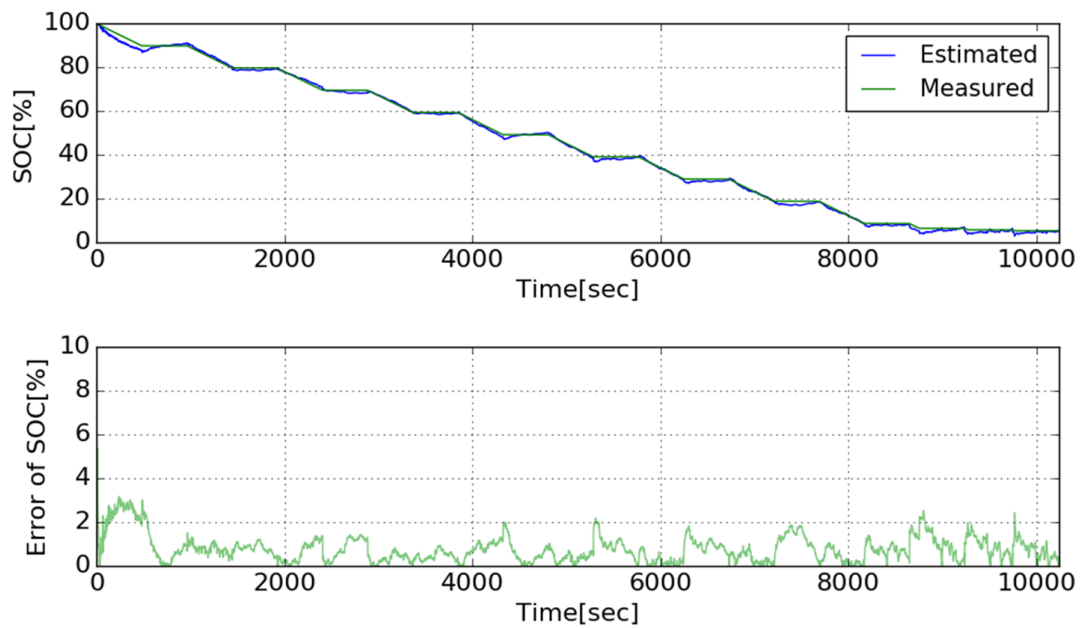


図 4-12 LSTM による推定誤差比較

Fig.4-12 Comparison of SOC estimations by LSTM using measured training data (LSTM24).

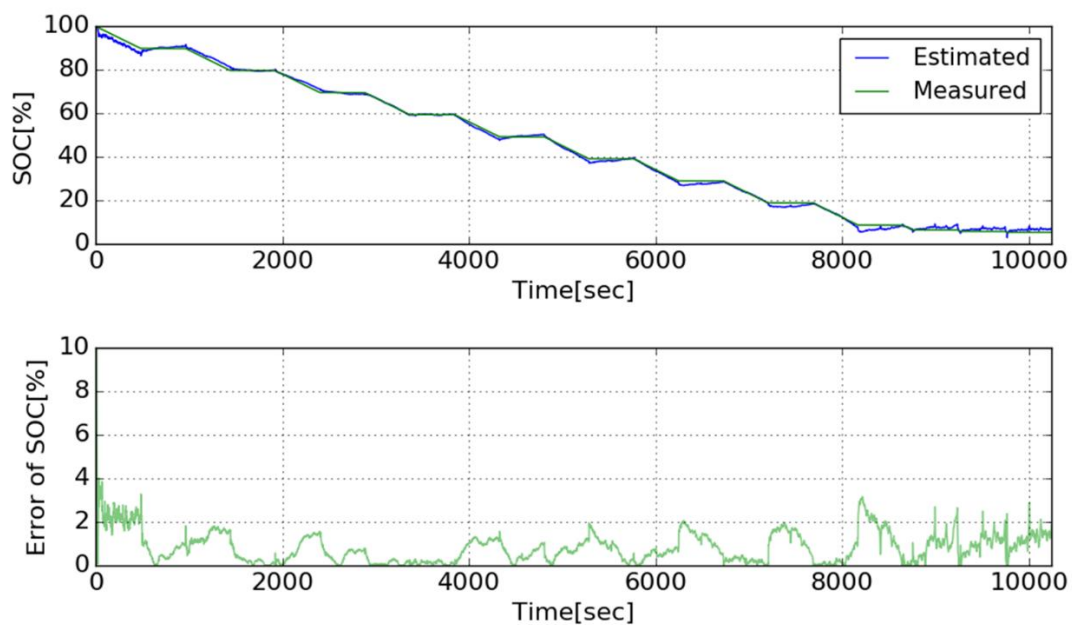


図 4-13 RLS+EKF 推定値を加えた推定誤差比較

Fig.4-13 Comparison of SOC estimations by using measured and RLS+EKF estimated SOC (LSTM+est).

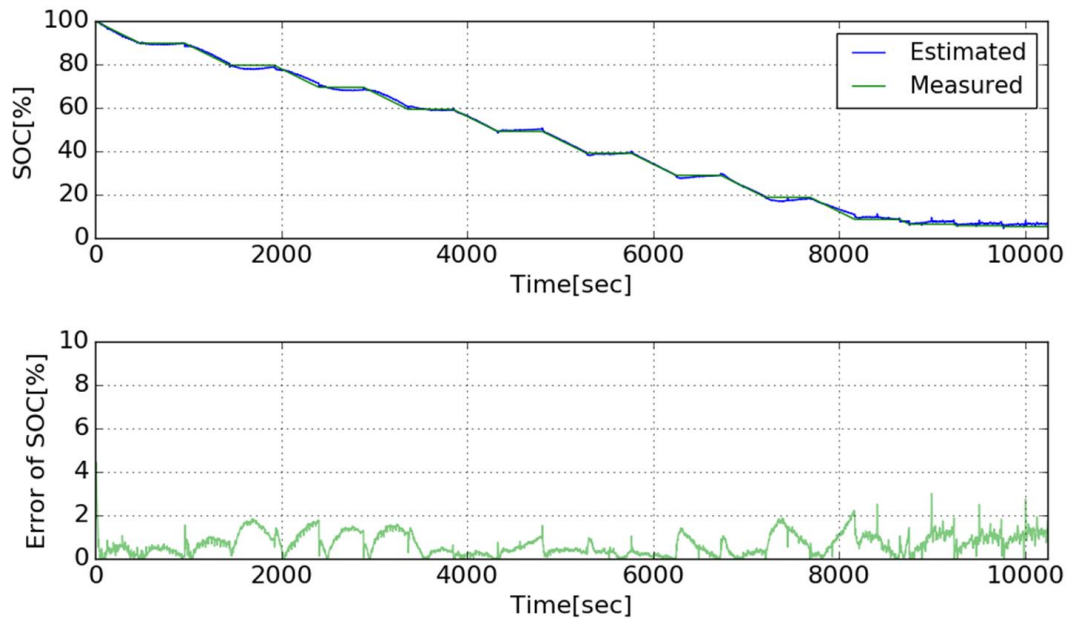


図 4-14 学習データ生成法と転移学習を組み合わせた場合の推定誤差比較

Fig.4-14 Comparison of SOC estimations by LSTM using measured and generated training data with TL (LSTM+est+TLb).

(3) 転移学習による学習の効率化

過去の知識モデルを活用する転移学習の利用方法を検討した。条件には異なる電池の知識モデル M_A 、同じ電池の知識モデル M_B があり、転移学習では通常の転移学習(TL)と学習範囲を大きくしたファインチューニング(FT)がある。 M_A は LIB A の 4 個のデータから LSTM で学習して作成した。 M_B は LIB B から 3 種類の放電パターン毎にランダムに 1 個のデータを選択し、 $3 \times 1 = 3$ 個を評価用データに設定した。残り 8 個のデータ $3 \times 8 = 24$ 個から、各放電パターンでランダムに 4 個のデータを選択し $4 \times 3 = 12$ 個のデータから LSTM で学習して M_B を作成し、残る $4 \times 3 = 12$ 個を学習用データとして使用した。転移学習せずに LSTM 学習したものを LSTM12, M_A で転移学習(TL)し LSTM 学習したものを LSTM+TLa, ファインチューニング(FT)し LSTM 学習したものを LSTM+FTa とし, M_B で転移学習(TL)し LSTM 学習したものを LSTM+TLb, ファインチューニング(FT)し LSTM 学習したものを LSTM+FTb として, M_A と M_B の違いにおける転移学習の精度を比較した結果を図 4-15 に示す。

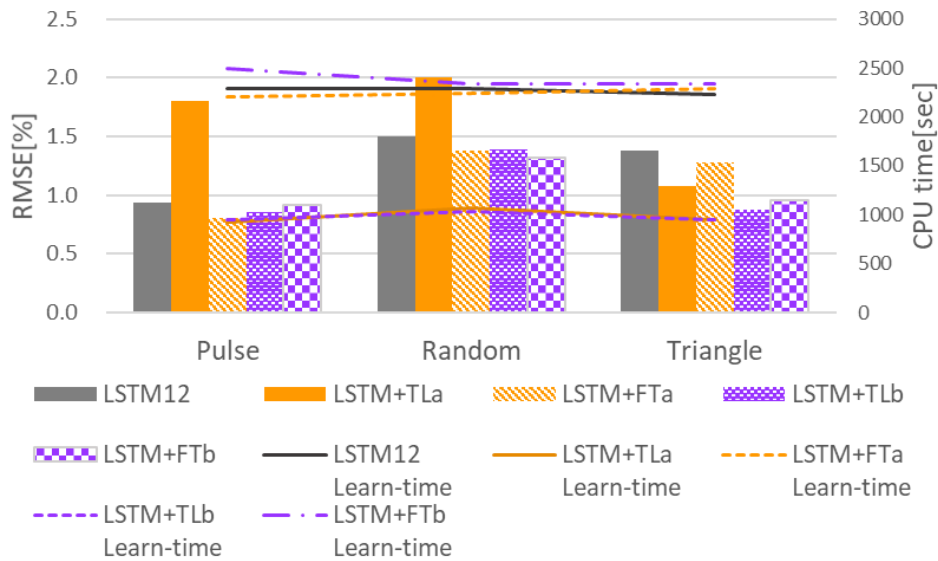


図 4-15 転移学習した知識モデルの違いによる推定精度

Fig.4-15 Comparison of SOC estimation using LSTM with transfer learning by difference in each learning model.

LSTM12 と比較して、 M_A の転移学習 LSTM+TLa ではパルス放電で RMSE 1.8%と精度が 2 倍悪化した。しかし、 M_A のファインチューニング(LSTM+FTa)では精度と計算時間ともにはほぼ同等の結果を得た。しかし、 M_B の転移学習(LSTM+TLb)では、転移学習する場合としない場合と比較すると、転移学習もファインチューニングも放電パターン別では RMSE 0.2%~0.4%の差の精度と、学習時間では約 2200 秒から約 1000 秒に短縮され 55%減の学習効果が確認できた。これらのことから、同じ電池での運用データから推定値をもとに教師データを構築し、これを学習して作成した知識モデルを活用することの有効性を確認できた。

本提案手法における実際の運用での利用方法について、図 4-16 にシーケンス図を示す。 Δt は再学習する周期を表す。BMSにおいて再学習により得られた予測モデルへ置き換える。

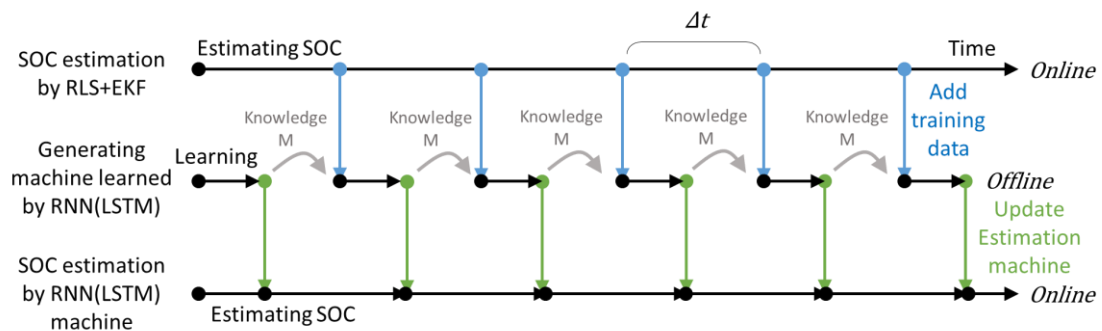


図 4-16 シーケンス図

Fig.4-16 Sequence diagram of this method.

以上のように、統計的推定手法を用いて教師データを構築する方法により、深層学習を活用した SOC 推定方法について、その要素技術の有効性を実験により確認することができた。

4.6 まとめ

ディープラーニングを使用するにはその特徴を表す適切な教師データを多数準備する必要があり、負担が大きい。本稿では、教師データとして実際の機器使用状態において観測可能な端子電圧や電流の観測データを利用することで偏りのない多様な条件下で多くのデータが準備できる方法を提案した。SOC の実測値と推定値の差の平均平方二乗誤差(RMSE) 評価指標で精度比較した。EKF 法では放電パターンの違いにより約 0.5~3.0%で精度にバラツキが生じるが、本手法はすべてのデータで約 1.0%前後の安定した精度を実験的に確認した。さらに過去に学習した知識モデルを活用した転移学習 (ファインチューニング) により教師データ数が半分でも精度 1.0%前後を維持し、学習時間では約 2200 秒から約 1000 秒に短縮され 55%減の転移学習の効果を確認できた。本方法により BMS からの観測可能な測定値を利用して教師データを構築することでロバスト性の高い SOC 推定が可能になり、同じ知識モデルを活用することで精度の維持向上が期待できる。これまで負担の大きかったデータ準備に対する有効な方法として期待できることが確認できた。

第5章 結論

本論文では、深層学習を活用して、LIBのSEM画像データから空隙箇所を自動抽出する作業の効率化と、BMSから入手した端子電圧と電流の測定値を用いて蓄電池内部の状態管理に必要なSOC推定の高精度化を目的とし、まとめたものである。

5.1 本研究の成果

深層学習の活用には、適切なネットワークモデルの構築、教師データの準備と、学習時間の短縮が大きな課題として認識される。本手法により教師データを準備し、構築したネットワークモデル（画像認識にはCNN、時系列データ処理にはRNN）を用いて同教師データを学習し作成した予測モデルにより、未来のデータ（評価用データ）に対して計算機実験から良好な結果が得られ、本手法の有効性を確認することができた。

第2章では、提案手法を説明する準備として、基本的なLIBの構造と動作原理について説明した。次に電池の状態や回路モデルの定義と、SOC推定方法に使用される代表的な電流積算法や、EKFなどの統計的手法やネットワークモデルを構築する上で必要な深層学習についての基本理論を説明した。

第3章では、U-Netモデルをベースにネットワークモデルを構築した。学習効率に悪影響を及ぼす内部共変量シフトを抑えるため特徴量マップにバッチノーマライゼーションを加え、畳み込み層数やフィルタ数、画像のならし効果やロバスト性向上のためプリーミング条件などを最適化した。さらに構成を左右対称にしてアップサンプリングで特徴マップをクロップせずそのまま利用した。これらの特徴を持つ独自のネットワークモデルを構築した。同モデルを使用してSEM画像データから空隙情報を自動抽出する方法を提案した。教師データの準備では、熟練者により画像データ合計1127枚の教師データを作成した。熟練者が作成した空隙抽出データと予測結果のピクセル単位の重なりからIoU (Intersection over Union)評価指標で精度比較した。被験者は初心者（入社2年）から熟練者（入社25年）までの材料開発者10名で構成する。経験の違いにより被験者間で精度に2~5倍の差が確認できたことから、空隙を見分けるには電池材料の立体的構造がイメージできる経験に裏付けされた判断が必要であると考えられる。被験者のIoU精度の平均値(mean) 30.1%に対して本手法は約1.5倍の44.4%で抽出した。最も高い精度49.2%の熟練者のレベルを越すことはできなかったが、かなり近い精度を実験的に確認できた。抽出作業は1画像当たり人手で1時間程度、本手法は数秒程度で完了した（学習時間除く）。また、本手法のIoU精度は、標準FCNモデル30.3%、FCN改モデル52.7%に比べ67.5%であり、明らかな優位性が見られた。以上のことから本手法による空隙抽出の自動化による作業効率向上への有効性が確認され工学的に有用であると考えられる。

第4章では、まず実験室レベルで諸条件を設定した等価回路モデルによって得られた実測値（端子電圧、電流、SOC）を真値として教師データに用いた。次に実際にBMS運用時を想定し、BMSから出力される端子電圧と電流の測定値を用いて、高速に推定処理が可能な統計的推定手法(RLS-EKF)によりSOC推定値を求めて、このデータ（端子電圧、電流、SOC）を教師データに追加し、再学習させた。運用時には実際の環境に合った教師データの準備が可能になった。このように教師データに実際の使用環境に基づく多様なデータを追加することで、教師データ構築方法の有効性を確認できた。これらの特徴を持つ革新的な独自の教師データ構築方法を提案した。次に学習時間の短縮には過去に学習した知識モデルを利用して転移学習することで、学習効率が向上することが実験により確認できた。SOCの実測値と推定値の差の平均平方二乗誤差(RMSE)評価指標で精度比較した。EKF法では放電パターンの違いにより約0.5~3.0%で精度にバラツキが生じるが、本手法はすべてのデータで約1.0%前後の安定した精度を実験的に確認した。さらに過去に学習した知識モデルを活用した転移学習（ファインチューニング）により教師データ数が半分でも精度1.0%前後を維持し、学習時間では約2200秒から約1000秒に短縮され55%減の転移学習の効果を確認できた。

実験により得られた成果は、本提案手法におけるネットワークモデルと教師データ準備方法と学習による効果であり、革新性と有効性の高さの可能性を示した。

5.2 今後の発展性

本論文で提案した、深層学習を活用して空隙箇所の抽出の自動化で用いたCNNによる画像認識技術と、電池内部の状態管理に必要なSOC推定の高精度化で用いたRNNによる時系列データ処理技術についての最適化方法と、教師データ構築方法を解明した。これら技術の今後の発展性について述べる。

深層学習を活用するには、多くの質の高い教師データを準備する必要がある。現状、SEM画像へ対象とする物体をマーキングして教師データを作成する作業に多くの時間を要している。しかも精度が求められるために熟練者によって教師データを作成する必要がある。別の電池であっても過去に学習した知識モデルがあれば、これを活用することで少ない教師データでも高い精度で物体検出できるようになれば、学習効率を向上させることができる。推定技術で転移学習などの知識モデルを活用する方法を解明したが、CNNでも同様にカスタマイズして最適化することで、高い抽出精度を得る方法を検討し、実験を通じてその有効性を確認する。たとえ教師データが少なくても学習効率を向上させることで熟練者並みの精度の物体検出が可能になり、電池材料開発を加速する一つの方法として貢献したいと考える。

次に推定技術の高精度化では、異なる電池特性の知識モデルで学習効率が向上することが実験で確認できたが、今回の評価実験で使用した電池は、どちらも正極材の材料として三元系リチウムを使用する三元系であった。例えばリン酸鉄リチウムを使用するリン酸鉄系

(LFP)の電池の場合には、リン酸鉄リチウムは放電中の電圧変化が小さく、容量が少なくなると急速に電圧降下する傾向にある。特に初期値誤差が大きい場合に精度が悪化する傾向にあり、高い精度でSOC推定するのが難しい。転移学習の中でもファインチューニングを利用することで、どれほど学習効果が期待できるか、電池種類の異なるLFPを用いてその都度モデルの最適化を行わずに、最小限の変更だけで精度の高いSOC推定が可能かを確かめる。

今後、これら推定技術を用いて電池の内部状態を推定し、将来の状態を予測し、さらにSEM画像データ解析で得た破壊検証による情報を学習した知識モデルと推定技術による非破壊検証による知識モデルを関連付け知識を融合することで、将来的に電池状態の異常検知や不具合予兆検知へと発展したいと考える。これら蓄電池から得られる様々な知識モデルをサーバに蓄積してビッグデータ情報として分析・活用することで、将来の電池の安全でかつ有効な利用に役立てることを願う。

参考文献

- [1] 経済産業省, カーボンニュートラルに伴うグリーン成長戦略, 成長戦略会議資料_2, 25 Dec, 2020.
- [2] 小久見善八, 西尾晃治, 図解革新型蓄電池のすべて, オーム社, 2011.
- [3] “Materials Genome Initiative, the Subcommittee of the Materials Genome Initiative, “<https://www.mgi.gov/>
- [4] “The NOMAD Laboratory, A European Centre of Excellence,” <https://NOMAD-coe.eu/>
- [5] Y. Wang, W.D. Richards, S.P. Ong, L.J. Miara, J.C. Kim, Y. Mo, and G. Ceder, “Design principles for solid-state lithium superionic conductors,” *Nature Materials*, Vol.14, pp.1026-1031, Oct. 2015.
- [6] Y. Mo, S.P. Ong, and G. Ceder, “First Principles Study of the Li₁₀GeP₂S₁₂ Lithium Super Ionic Conductor Material,” *Chem. of Mat.*, 24 Issue 1, pp.15-17, Jan. 2012.
- [7] 内閣府, 戦略的イノベーション創造プログラム(SIP), SIP 第 1 期研究開発概要(11 課題),” <https://www8.cao.go.jp/cstp/gaiyo/sip/>, 参照 Dec.4, 2021.
- [8] 福満仁志, 寺田健二, 末広省吾, 滝克彦, 千容星, “三次元空隙ネットワーク解析リチウムイオン電池電極の評価法の開発,” *Electrochemistry*, 83 巻 1 号, pp.2-6, Jan. 2015.
- [9] P. Weicker, A Systems Approach to Lithium-Ion Battery Management, *Artech House*, 2014.
- [10] 一般社団法人電池工業会, “電池について>規格・サイズ・構造>電池の構造と反応式(例)リチウムイオン二次電池,” <https://www.baj.or.jp/battery/knowledge/structure.html>, 参照 Nov.23, 2021.
- [11] 堀江明英, リチウムイオン電池「基礎と応用」, 培風館, 東京, 2010.
- [12] G. Sikha, B.N. Popov, and R.E. White, “Effect of Porosity on the Capacity Fade of a Lithium-Ion Battery: Theory,” *Journal of the Electrochemical Society*, vol.151, no.7, pp.A1104-A1114, 2004.
- [13] H. Zheng, G. Liu, X. Song, P. Ridgway, S. Xun, and V.S. Battaglia, “Cathode Performance as a Function of Inactive Material and Void Fractions,” *Journal of the Electrochemical Society*, vol.157, no.10, pp.A1060, 2010.
- [14] D. Kehrwald, P.R. Shearing, N.P. Brandon, P.K. Shinha, and S.J. Harris, “Local Tortuosity Inhomogeneities in a Lithium Battery Composite Electrode,” *Journal of the Electrochemical Society*, vol.158, no.12, pp.A1393, 2011.
- [15] 横山豊和, 井上義之, “電池の開発と製造技術向上のための粉体プロセス新展開ーリチウムイオン二次電池の高機能化に向けた粒子設計加工技術ー,” 粉砕 <特集 I> ナノパーティクルテクノロジーの構築と実用化への展開, no.55, pp.38-47, 2012.

- [16] 福満仁志, 寺田健二, 末広省吾, 滝克彦, 千容星, “三次元空隙ネットワーク解析によるリチウムイオン電池電極の評価法の開発,” *Electrochemistry*, vol.1, no.83, pp.2-6, 2015.
- [17] 足立修一, 廣田幸嗣, 押上勝憲, 馬場厚志, 丸田一郎, 三原輝儀, バッテリマネジメント工学, 東京電機大学出版局, 東京, 2015.
- [18] G.L. Plett, “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1. Background,” *Journal of Power Sources*, vol.134, no.2, pp.252-261, 2004.
- [19] G. Wu, R. Lu, C. Zhu, and C.C. Chan, “State of charge estimation for NiMH battery based on electromotive force method,” in *Proc. VPPC*, pp.1-5, 2008.
- [20] G.L. Plett, “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs Part 2. Modeling and identification,” *Journal of Power Sources*, vol.134, no.2, pp.262-276, 2004.
- [21] M. Oya, W. Sueki, Y. Hayakawa, K. Takaba, and M. Fukui, “Combined RLS-EKF method for simultaneous SOC and parameter estimations for lithium-ion batteries,” *ECS Transaction*, vol.80, no.10, pp.207-217, 2017.
- [22] 林磊, 有馬理仁, 福井正博, “リチウムイオン蓄電池の同時かつ安定的な回路パラメータと残量の推定法,” 信学論 (B) , Vol.J104-B No.3, pp.222-231, 2021.
- [23] E. Chemali, P.J. Kollmeyer, M. Preindl, and A. Emadi, “State-of-charge estimation of Li-ion batteries using deep neural networks: A machine learning approach,” *Journal of Power Sources*, vol.400, pp.242-255, 2018.
- [24] E. Chemali, P.J. Kollmeyer, M. Preindl, R.Ahmed, and A. Emadi, “Long Short-Term Memory Networks for Accurate State-of-Charge Estimation of Li-ion Batteries,” *IEEE Trans. Industrial Electronics*, vol.65, no.8, pp.6730-6739, 2018.
- [25] S. Bockrath, R. Roskopf, S. Koffel, S. Waldhör, K. Strivastava, and V.R.H. Lorentz, “State of Charge Estimation using Recurrent Neural Networks with Long Short-Term Memory for Lithium-Ion Batteries,” *IECON 2019 45th Annual Conf. of the IEEE Industrial Electronics Society*, pp.2507-2511, 2019.
- [26] G.L. Plett, “Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs Part 3. State and Parameter Estimation,” *Journal of Power Sources*, vol.134, no.2, pp.277-292, 2004.
- [27] G.L. Plett, “Sigma point Kalman filtering for battery management systems of LiPB-based HEV battery packs Part 1. Introduction and State Estimation,” *Journal of Power Sources*, vol.161, no.2, pp. 1356-1368, 2006.
- [28] G.L. Plett, “Sigma point Kalman filtering for battery management systems of LiPB-based HEV battery packs Part 2. Simultaneous State and Parameter Estimation,” *Journal of Power Sources*, vol.161, no.2, pp. 1369-1384, 2006.

- [29] R.K. Mehra, "On the Identification of Variance and Adaptive Kalman Filtering," *IEEE Trans. Automatic Control*, vol.AC-15, pp.175-184, 1970.
- [30] R.K. Mehra, "On-line Identification of Linear Dynamic Systems with Applications to Kalman Filtering," *IEEE Trans. Automatic Control*, vol.AC-16, pp.12-21, 1971.
- [31] A.H. Mohamed and K.P. Schwarz, "Adaptive Kalman Filtering for ING/GPS," *Journal of Geodesy*, vol.73, pp.193-203, 1999.
- [32] C. Vidal, P. Kollmeyer, E. Chemali, and A. Emadi, "Li-ion Battery State of Charge Estimation using Long Short-Term Memory Recurrent Neural Network with Transfer Learning," *ITEC 2019 IEEE Transportation Electrification Conference and Expo*, 2019.
- [33] 総務省, ICT (情報通信技術) を学ぶ 総務省 ICT スキル総合習得プログラム, 総務省 情報活用支援室, https://www.soumu.go.jp/ict_skill/, 参照 Nov. 16, 2021.
- [34] S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd Edition, *Prentice Hall*, 2010.
- [35] 巢籠悠輔, 詳細ディープラーニング (第2版), マイナビ出版, 東京, 2019.
- [36] D.P. Kingma and J. Ba, "ADAM: A Method for Stochastic Optimization," *ICLR2015 International Conference on Learning Representations*, San Diego, 2015.
- [37] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," *In Proc. of the 32nd International Conf. on Machine Learning*, Mar. 2015.
- [38] K. Simonyan, A. Vedaldi, and A. Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps," *In ICLR 2014 Workshop Track*, Apr. 2014.
- [39] H. Lee, R. Grosse, R. Ranganath, and A.Y. Ng, "Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations," *In proc. of the 26th Annual International Conf. on Machine Learning, ACM*, pp.609-616, Canada, Jun. 2009.
- [40] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *NIPS'12 Proc. of the 25th International Conf. on Neural Information Processing Systems*, Vol.1, pp.1097-1105, Dec. 2012.
- [41] J. Long, E. Shelhamer, and T. Darrel, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Conf. on Computer Vision and Pattern Recognition*, pp.3431-3440, Oct. 2015.
- [42] H. Noh, S. Hong, and B. Han, "Learning Deconvolution Network for Semantic Segmentation," *ICCV'15 Proc. of the 2015 IEEE International Conf. on Computer Vision*, pp.1520-1528, Dec. 2015.

- [43] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.39, Issue 12, pp.2481-2495, Dec. 2017.
- [44] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Medical Image Computing and Computer-Assisted Intervention, MICCAI*, pp.234-241, May 2015.
- [45] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation," *Computer Vision and Pattern Recognition*, Nov. 2016.
- [46] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," *Computer Vision and Pattern Recognition*, Apr. 2017.
- [47] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A Nested U-Net Architecture for Medical Image Segmentation," *Computer Vision and Pattern Recognition*, Jul. 2018.
- [48] V. Zyuzin and T. Chumarnaya, "Comparison of Unet architectures for segmentation of the left ventricle endocardial border on two-dimensional ultrasound images," *Ural Symposium on Biomedical Engineering*, Apr. 2019.
- [49] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol.9, no.8, pp.1735-1780, 1997.
- [50] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *arXiv:1412.3555*, 2014.
- [51] M. Schuster and K.K. Paliwal, "Bidirectional Recurrent Neural Networks," *IEEE Trans. Signal Processing*, vol.45, no.11, pp.2673-2681, 1997.
- [52] B. Dzmitry, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *arXiv:1409.0473*, 2014.
- [53] J. Bradbury, S. Merity, C Xiong, and R. Socher, "Quasi-Recurrent Neural Networks," *ICLR2017 International Conference on Learning Representations*.
- [54] S. J. Pan and Q. Yang, "A Survey on Transfer Learning," *IEEE Trans. Knowledge and Data Engineering*, Vol.22, No.10, pp.1345-1359, 2009.
- [55] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A Survey on Deep Transfer Learning," *ICANN2018 International Conference on Artificial Neural Networks*.
- [56] A.P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognition*, Vol.30, Issue 7, pp.1145-1159, Jul. 1997.
- [57] M. Everingham, S.M.A. Eslami, L.V. Gool, C.K.I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge: A Retrospective," *International Journal of Computer Vision*, 111, pp.98-136, Jan. 2015.

- [58] N. Wassiliadis, T. Herrmann, L. Wildfeuer, C. Reiter, and M. Lienkamp, “Comparative Study of State-of-Charge Estimation with Recurrent Neural Networks,” *IEEE Transportation Electrification Conference and Expo (ITEC)*, 19-21 Jun. 2019.
- [59] M. Charkhgard and M. Farrokhi, “State-of-Charge Estimation for Lithium-Ion Batteries Using Neural Networks and EKF,” *IEEE Trans. on Industrial Electronics*, vol.57, issue 12, pp.4178-4187 Dec. 2010.
- [60] C. Kun, M. Zhiwei, L. Yuehua, J. Zhinong, and Z. Jinjie, “Lithium-ion battery state of charge estimation based on dynamic neural network and Kalman filter,” *IEEE International Conference on Prognostics and Health Management (ICPHM)*, 11-13 Jun. 2018.
- [61] F. Yang, X. Song, F. Xu, and K.L. Tsui, “State-of-Charge Estimation of Lithium-Ion Batteries via Long Short-Term Memory Network,” *IEEE Access: Topic: Advances in Prognostics and System Health Management*, vol.7, 23 Apr. 2019.
- [62] Antonio Gulli, Sujit Pal, 直感 Deep Learning, オライリー・ジャパン, 東京, 2018.
- [63] Panasonic Corporation, “電子デバイス・産業用機器>商品一覧>産業用電池>二次電池>リチウムイオン電池>品番>UR18650AA : リチウムイオン電池・特性値” <https://industrial.panasonic.com/jp/products/pt/lithium-ion/models/UR18650AA>, 参照 Nov.23, 2021.
- [64] Panasonic Corporation, “電子デバイス・産業用機器>商品一覧>産業用電池>二次電池>リチウムイオン電池>品番>NCR18650BF : リチウムイオン電池・特性値” <https://industrial.panasonic.com/jp/products/pt/lithium-ion/models/NCR18650BF>, 参照 Nov.23, 2021.

謝辞

本研究をまとめるにあたり、懇切丁寧なご指導を賜りました立命館大学 大学院理工学研究科 福井正博教授に心より感謝申し上げます。福井先生には松下電器産業にご所属の時より直接、間接にご指導頂き、私の研究者としての礎を築くことに対して大変お世話になりました。また近年では未来を支える技術の 1 つであるリチウムイオン蓄電池の可能性へと導いていただき、リチウムイオン蓄電池に係る数理解析とモデリングの技術、さらにディープラーニングなどのコンピュータサイエンスを適用する技術をご教授頂き、また博士論文審査の主査をして頂きました。ここに改めてお礼申し上げます。

立命館大学 大学院理工学研究科 鷹羽浄嗣教授、中山良平教授には博士論文審査の副査をして頂きました。ここにしてお礼申し上げます。

本研究を進める中で多大なるご指導とご意見を頂き、さらに多くの議論をさせて頂きました元立命館グローバル・イノベーション研究機構の林磊博士に深く感謝申し上げます。

また2021年3月まで本研究室に在籍されておりました、大和製罐株式会社 有馬理仁博士にもたくさんの議論を交わし、刺激を与えて頂き、アドバイスを頂いたこと深く感謝申し上げます。

TDK株式会社 内野様にはミュンヘン出向時から電池システムに関する議論やアドバイスを頂いたこと、現在も同僚としてご支援を頂きましたことに深く感謝申し上げます。

立命館大学大学院理工学研究科 福井研究室の松下修平様、磯田有哉様、さらに元修士課程学生の常田さん、川瀬さん、修士課程・学部の学生の皆様に感謝申し上げます。

本研究をするにあたり多大なるご理解とご配慮を賜りましたTDK株式会社 中野敦之様、そして同僚の皆様に感謝申し上げます。

最後に、私の夢を理解し長い間応援し、常に暖かく見守ってくれた妻 晴美ならびに家族に感謝します。

塩原 孝弘