

手書き医療用語認識における 有効的なデータ拡張手法の検討

新堂 風

令和2年2月

電気情報工学科

概要

Portable Health Clinic（以下，PHC）は発展途上国農村部における健康促進に向けた遠隔医療システムである。ヘルスアシスタントと呼ばれるスタッフが複数の健康測定器具を医者のない農村部に持ち込み、村民に対して健康診断を行う。健康診断の結果、医者からの診断が必要であると判断された患者は都市部にいる医者と電話を通して繋がり、医者が直接診断できない発展途上国農村部においても人々は診断を受けることができる。このシステムでは、医者は患者を診断しながら症状や処方薬などをノートに取り、通話後にそれをコンピュータに入力して処方箋を作成する。このときにノートに書かれた手書き文字を認識し、その情報を元に処方箋を予測してコンピュータに入力する手間を削減できれば、医者の時間を節約でき、医者はさらに多くの人々の診断を行うことができる。

現在医療用語に特化したデータセットがオープンソースとして存在しないため、PHCにおける過去の処方箋データから頻出する単語を座標の時系列データとして収集する。様々な手書き文字に対応するため、本研究では多様な提供者から大量のデータを得る必要があるが、十分なデータ量を確保するためには非常に多くの労力と時間を要する。十分な量のデータ収集なしに高精度の文字認識ができるオンライン手書き医療用語認識の実現を

目指して、現在データ拡張手法を用いたオンライン手書き医療用語認識 [1] が開発されている。先行研究 [1] は、ストロークの回転と平行移動によってデータ量を水増しする Stroke Rotation and Parallel-shift (SRP) 手法を用いている。この用語認識はデータ量が少ない状況での学習を実現でき、高精度でのオンライン手書き医療用語認識が可能であることが確認されている。しかし SRP 手法において回転の大きさ、平行移動の大きさは筆者が文字の形を崩さないと判断できる範囲で設定しているため、データを高倍率で拡張した場合に同じような値を用いてしまい、拡張後に同じようなデータが増えて学習精度が低くなる原因になってしまう。そのためこの用語認識は学習の際にランダムに設定された回転の大きさと平行移動の大きさの組み合わせにより、認識精度にばらつきが出てしまうという問題がある。

本論文では、処方箋の予測に向けたシステムの初期研究として、安定して高精度で学習することができるオンライン手書き医療用語認識システムを提案する。またオンライン手書き文字のデータ拡張手法として文字の縦横比を変更しデータ量を水増しする Ratio 手法を新たに提案する。Ratio 手法は文字の縦横比を変更するため、拡張後に文字のストロークが重なることなく文字の形を大きく変えてしまうことがないままデータの多様化を行うことができる。本論文では既存の SRP 手法と Ratio 手法を組み合わせデータ拡張を実現した。収集した 15991 語のデータを用いて、SRP 手法と Ratio 手法でデータ拡張を行った後に BidirectionalLSTM で学習を行った結果、480 語のクラスにおいて平均精度 93.0%、最小精度 92.1% で単語を認識した。この結果は SRP 手法のみの場合の認識精度と比べて平均精度は 3.5%、最小精度は 14.7% 向上した。データ拡張を行わなかった場合の認識精度と比べて平均精度は 19.6%、最小精度は 73.1% 向上した。

目次

第1章	はじめに	1
1.1	研究背景	1
1.2	研究目的	2
1.3	論文構成	2
第2章	文字認識に関する関連技術	4
2.1	手書き文字認識の種類	4
2.2	再帰型ニューラルネットワーク	5
2.2.1	RNNの構造	6
2.2.2	LSTM	8
2.3	関連研究	9
2.3.1	直線データと BidirectionalLSTM を用いた中国語認識システム [2]	9
2.3.2	dropStroke 手法を用いたデータ拡張と筆者同定システム [3]	9
2.4	先行研究	10
第3章	手書き医療用語認識と SRP 手法・Ratio 手法によるデータ拡張	11
3.1	システムの概要	11
3.2	前処理ブロック	12

3.2.1	余分な点の除去	13
3.2.2	正規化	13
3.2.3	特徴量抽出	14
3.3	データ拡張ブロック	15
3.3.1	SRP 手法	15
	ストロークの回転	15
	ストロークの平行移動	16
3.3.2	Ratio 手法	18
3.4	機械学習ブロック	20
第 4 章	実装	23
4.1	データ収集	23
4.2	使用機器・学習モデル構造	24
第 5 章	評価	27
5.1	各値の設定	27
5.2	評価方法	29
5.3	評価結果	29
第 6 章	おわりに	32
6.1	本研究の主たる成果	32
6.2	今後の課題	33
	謝辞	34
	参考文献	35

第1章

はじめに

1.1 研究背景

Portable Health Clinic（以下，PHC）は発展途上国農村部における，健康促進のための遠隔医療システムである [4]．ヘルスアシスタントと呼ばれるスタッフが複数の健康測定器具を医者のない農村部に持ち込み，村民に対して健康診断を行う．健康診断の結果，医者からの診断が必要であると判断された患者は都市部にいる医者と電話を通して繋がり，診断を受けることができる．このシステムによって，医者が直接診断できない発展途上国農村部においても人々は診断を受けることができる．このシステムでは，医者は患者を診断しながら症状や処方薬などをノートに取り，通話後にそれをコンピュータに入力して処方箋を作成する．このときにノートに書かれた手書き文字を認識し，その情報を元に処方箋を予測してコンピュータに入力する手間を削減できれば，医者の時間を節約でき，医者はさらに多くの人々の診断を行うことができる．

1.2 研究目的

本研究の目的は、遠隔医療における処方箋予測に向けたオンライン文字認識に適したデータの拡張方法を確立することである。オープンソースのオンライン文字認識用データセットのうち、医療用語に特化したものは存在していないため、先行研究 [1] では独自にデータ収集を行っている。しかしデータの収集には多くの時間と労力を要するため、先行研究ではストロークの回転と平行移動によってデータ量を水増しする SRP 手法を行ってデータ量を水増している。だが SRP 手法において回転の大きさ、平行移動の大きさは筆者が文字の形を崩さないと判断できる範囲で設定しており、データを高倍率で拡張した場合同じようなパラメーターが増えてしまい、拡張後に同じようなデータが増えて学習精度が低くなる原因になってしまう。そのため、この用語認識は学習の際にランダムに設定された回転の大きさと平行移動の大きさの組み合わせにより、認識精度にばらつきが出てしまうという問題がある。

現在、オンライン文字認識の研究においてデータ拡張を行っているものは非常に少なく、安定して高精度で認識できるオンライン文字認識の拡張方法は未だ確立されていない。機械学習においてデータ量は精度を大きく左右する重要なパラメータである。オフライン文字認識については多くのデータ拡張方法が研究されているが、オンライン文字認識についてもデータの拡張方法について議論がなされるべきである。

1.3 論文構成

本論文の構成は以下の通りである。第 2 章では、筆者らが目指しているオンライン手書き医療用語認識、及び本研究に至った取り組むべき課題について説明を行い、既存の対策の問題点をあげる。第 3 章では、問題解決に向け本研究で提案する手法について説明を行う。第 4 章では提案手法の実装について述べる。第 5 章では提案手法の評価につ

いて述べる．最後に第 6 章で本論文のまとめと今後の展望を述べる．

第2章

文字認識に関する関連技術

本章では、手書き文字認識技術として一般に用いられるオンライン文字認識とオフライン文字認識について説明する。さらに時系列データに対応した機械学習モデルである、再帰型ニューラルネットワークについて説明する。その後再帰型ニューラルネットワークを用いたオンライン文字認識の既存研究を述べ、遠隔医療での処方箋予測に向けたオンライン文字認識におけるデータ拡張に関する先行研究について述べる。

2.1 手書き文字認識の種類

手書き文字の認識は大きく2つに分けることができる。オフライン文字認識とオンライン文字認識である。オフライン文字認識は、文字の画像データにおいてそれぞれのピクセルが持つ情報を特徴量として文字を認識する技術であり、手書き文字は認識されるためにイメージスキャナやデジタルカメラによって読みとられる。オンライン文字認識は手書き文字の (x, y) 座標やスピード、筆圧などを特徴量として認識する技術で、通常タブレットなどに書き込まれた文字が認識される。

オフライン文字認識は画像認識技術であるため、文献 [5] のような畳み込みニューラ

ルネットワーク (Convolutional Neural Network, 以下 CNN) を用いた文字認識が多く研究されている。利用可能な画像データもインターネット上に多く存在するが、実際に認識を行う際には手書きされた文字をスキャナやデジタルカメラなどで読み取る必要があるなどのデメリットもある。

一方でオンライン文字認識は IAM On-Line Handwriting Database [6], CASIA Chinese Handwriting Database [7] などのデータセットが存在するが、オフライン文字認識に比べると利用可能なデータは少ない。しかし認識時においてはタブレットなどのデバイスに書き込まれた手書き文字のデータが直接使われるため、オフライン文字認識と比べると手間が少ない。また、オンライン文字認識は筆順・筆圧・書き込みにかかる時間などの、オフライン文字認識では使うことができない情報を用いて認識を行うことができるなどのメリットがある。

本研究では、医者の手書き文字は筆記体が多く使われるため画像での認識は難しいと考えられること、実際に使用する際にはスキャンやデジタルカメラによる撮影を行わずリアルタイムでの認識を想定していることから、オンライン文字認識を用いて医者の手書き医療用語の認識を行う。

2.2 再帰型ニューラルネットワーク

再帰型ニューラルネットワーク (Recurrent Neural Network, 以下 RNN) とは中間層に戻り値のある、音声、動画、文章などの時系列データの扱いに優れているニューラルネットワークである。以下で RNN の構造と、再帰型ニューラルネットワークの一種である Long Short-Term Memory (以下, LSTM) について説明する。本論文では LSTM を用いて機械学習を行う。

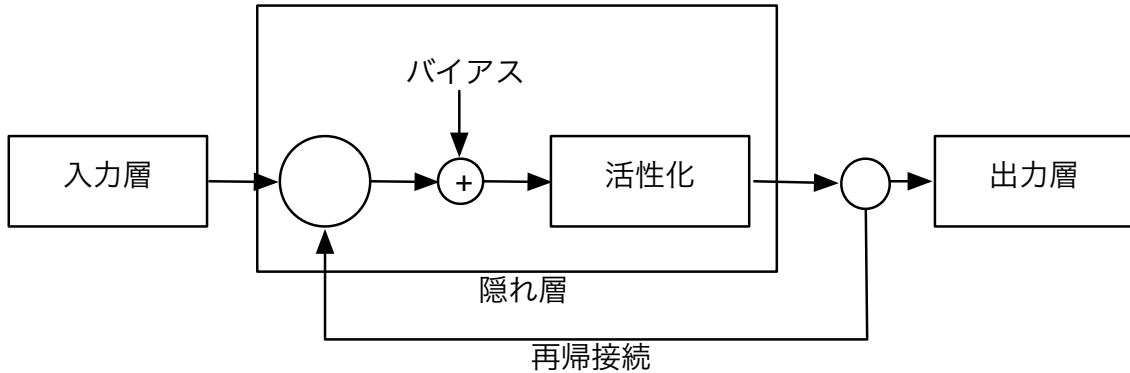


図 2.1: RNN の構造

2.2.1 RNN の構造

図 2.1 に再帰型ニューラルネットワークの構造を示す．RNN は中間層のノードごとに戻り値があるため，現在入力されているデータより前のデータの影響を考慮して計算を行うことができる．そのため RNN は連続的な情報を保持したまま学習を行うことができる [8]．式 2.1, 式 2.2 に，RNN がそれぞれのノードにおいて行っている計算を入力値 x と時間的順序 t を用いて示す．

$$H(t) = h(W_H x(t) + W_{self} H(t-1) + b_H) \quad (2.1)$$

$$Y(t) = y(W_Y H(t) + b_Y) \quad (2.2)$$

ここにおいて $H(t)$ は隠れ層の出力であり， W_H は隠れ層への入力の重み， W_{self} は戻り値の重み， b_H は隠れ層へのバイアス， $Y(t)$ はネットワークの出力， W_Y は隠れ層の出力の重み， b_Y は出力へのバイアス， $y()$ と $h()$ はそれぞれ出力層と隠れ層の活性化関数である．式より，隠れ層の出力は隠れ層への入力 $x(t)$ だけでなく，隠れ層からの 1 つ前の出力 $H(t-1)$ の影響も受けていることがわかる．したがってネットワークの出力

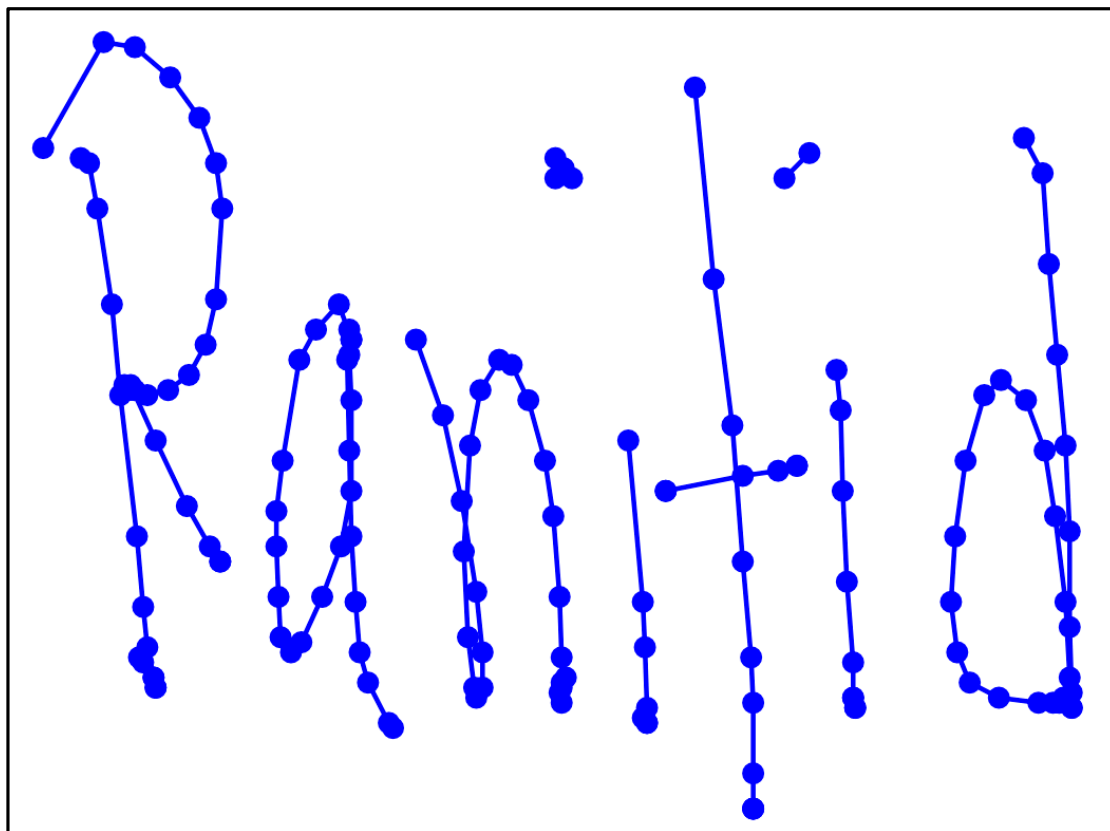


図 2.2: 文字を点で表した図

$Y(t)$ は $x(t)$ と $x(t-1)$ の影響を受けていると言える．この計算は最初をのぞいた全ての入力において行われている．

図 2.2 に認識する手書き文字の例を示す．このように，手書き文字は検出された座標が時間的順序に従って並んだものであると言える．オフライン文字認識では，手書き文字を画像データとして捉えるため，点の時間的順序を考慮に入れずに認識を行う．RNN の入力に連続した手書き文字の点データを用いることで，オンライン文字認識ではその時間的順序を考慮に入れて認識を行うことができる．

しかし，実際に RNN で出力に反映できる過去の入力情報は短く，時系列 10 ステッ

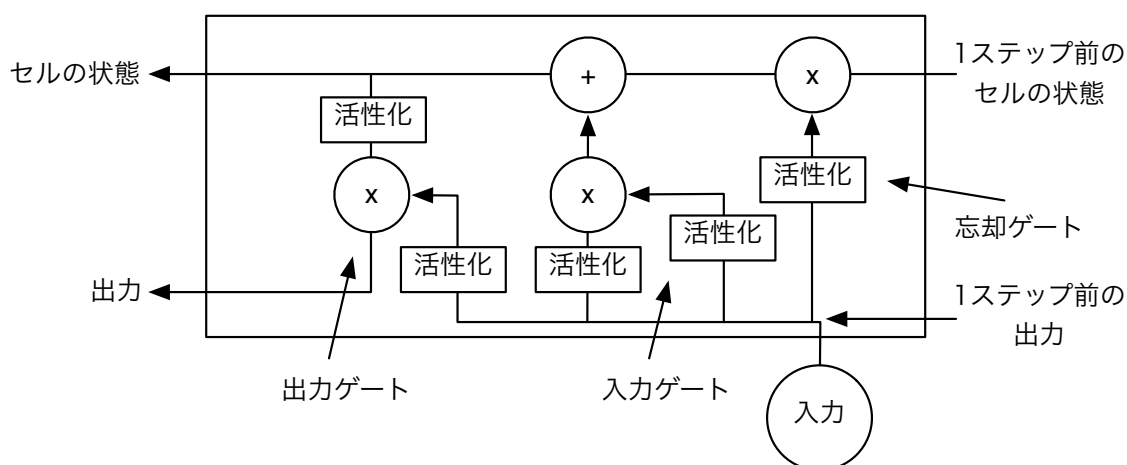


図 2.3: LSTM ブロックの構造

プ分程度であると言われている [9].

2.2.2 LSTM

図 2.3 に LSTM ブロックの構造を示す. LSTM は RNN よりも長期にわたる記憶を実現するための方法のひとつで, RNN の隠れ層の各ユニットを LSTM ブロックに置き換えたものである. RNN のユニットと異なり, LSTM ブロックは記憶を持ったセルの構造をしている. ゲートと呼ばれる特殊な構造がセルに情報を与えるかセルの情報を除去するかを決める.

忘却ゲートがセル内の情報をリセットし, 入力ゲートは, 入力が現在のセルにどれだけの影響を与えるか決める. 出力ゲートは出力が残りのネットワークにどれだけの影響を与えるか決める. この方法によって LSTM はより長い時系列データにおける再帰型ニューラルネットワークの利用を可能にしている.

2.3 関連研究

ここでは LSTM を用いたオンライン文字認識の既存研究と、文字認識におけるデータ拡張に関する既存研究について述べる。

2.3.1 直線データと BidirectionalLSTM を用いた中国語認識システム [2]

文献 [2] では中国語手書き漢字の座標データの中から、直線上の点や近接する点など、取り除いても文字として成り立つ点を除去したのち直線データに変換し、BidirectionalLSTM を用いて認識を行っている。BidirectionalLSTM については 3.4 節で述べる。点の除去を行うことで入力データを簡易化し、さらに直線データへの変更を行うことで連続する点同士の関係性（点同士の距離、次の点への角度、同一字画上にあるかなど）を特徴量として抽出することができる。機械学習モデルには BidirectionalLSTM を用いることで、入力されているより過去の座標の情報だけでなく未来の座標の情報も考慮に入れた学習を行うことができる。

2.3.2 dropStroke 手法を用いたデータ拡張と筆者同定システム [3]

文献 [3] では中国語手書き漢字において、座標の時系列データから一部のストロークを除去する処理を複数回繰り返すことでデータ拡張を行い、CNN を用いたオフライン認識で筆者の同定を行っている。文字としては不完全なデータになるが筆者の特徴を大きく変えることにはならず、100% に近い精度での筆者同定を可能にしている。ただし、ストロークを除去することで異なる文字・単語になってしまう可能性があり、本論文の目的とする単語認識においてこの手法は適切であるとは言えない。同著者から

は、筆者同定のためのデータ拡張として dropSegment 手法というものも提案されているが [10], これも異なる文字・単語に変わってしまう可能性があり, 単語認識に適切であるとは言えない.

2.4 先行研究

遠隔医療システムにおける処方箋予測に向けた手書き医療用語認識に関する研究 [1] では, 処方箋の予測に向けたシステムの初期研究として, 再帰型ニューラルネットワークの一種である BidirectionalLSTM を用いたオンライン手書き医療用語認識手法の提案を行っている. また様々な手書き文字に対応するため, 文字認識では多様な提供者から大量のデータを得る必要がある. しかし, 十分なデータ量を確保するためには非常に多くの労力と時間を要する. そこで筆者はオンライン手書き文字のデータ拡張手法として, ストロークの回転と平行移動によってデータ量を水増しする Stroke Rotation and Parallel-shift (SRP) 手法を提案している. 手書き文字データは, 文献 [2] を用いてデータの簡易化, 特徴量の抽出を行っている. また文献 [1] では Portable Health Clinic における過去の処方箋データから頻出する単語を座標の時系列データとして収集している. 収集した 15991 語のデータを用いて, SRP 手法でデータ拡張を行った後に BidirectionalLSTM で学習を行った結果, 480 語のクラスにおいて 89.5% の精度で単語を認識している. この結果はデータ拡張を行わなかった場合の認識精度と比べて 16.1% 高い. しかし SRP 手法において回転の大きさ, 平行移動の大きさは筆者が文字の形を崩さないと判断できる範囲で設定しているため, データを高倍率で拡張した場合に同じような値を用いてしまい, 拡張後に同じようなデータが増えて学習精度が低くなる原因になってしまう. そのため, この用語認識は学習の際にランダムに設定された回転の大きさと平行移動の大きさの組み合わせにより, 認識精度にばらつきが出てしまうという問題がある.

第3章

手書き医療用語認識と SRP 手法・Ratio 手法によるデータ拡張

本章では，先行研究 [1] の BidirectionalLSTM を用いたオンライン手書き医療用語認識を説明する．また，オンライン文字認識用のデータ拡張手法として先行研究の SRP 手法を説明し，データ拡張の新手法として Ratio 手法を提案する．

3.1 システムの概要

図 3.1 にシステムの概要を示す．提案システムでは，タブレットから収集した手書き医療用語の座標データに対して前処理を行う．その後学習プロセスでは前処理後のデータに拡張手法を適用し，機械学習における学習データとする．推定プロセスでは，学習を行ったモデルに前処理後のデータを入力し，用語を推定する．ニューラルネットワーク構造とデータ前処理，特徴量抽出に関しては 2.3.1 項の文献 [1] を参考にした．以下，各手順について述べる．

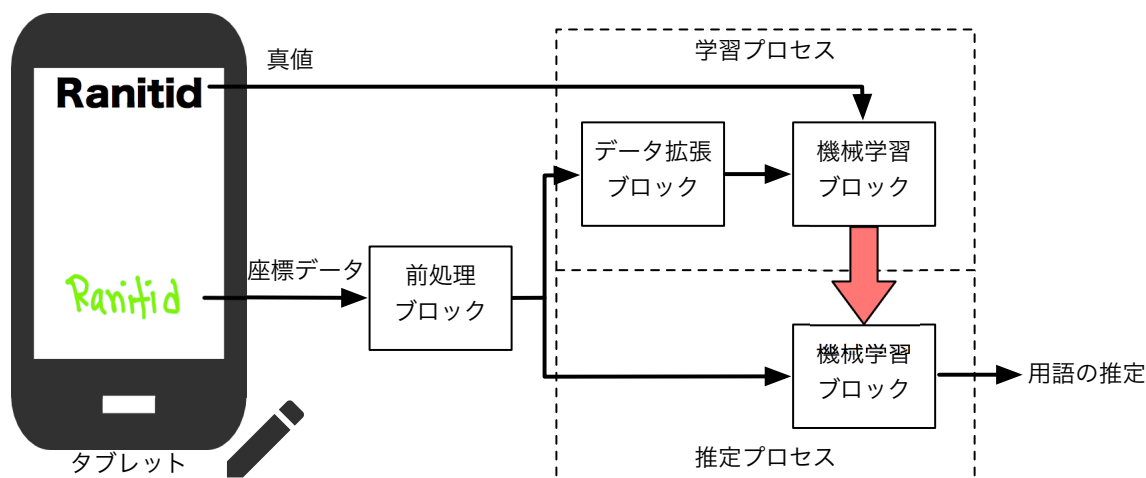


図 3.1: 提案システムの概要

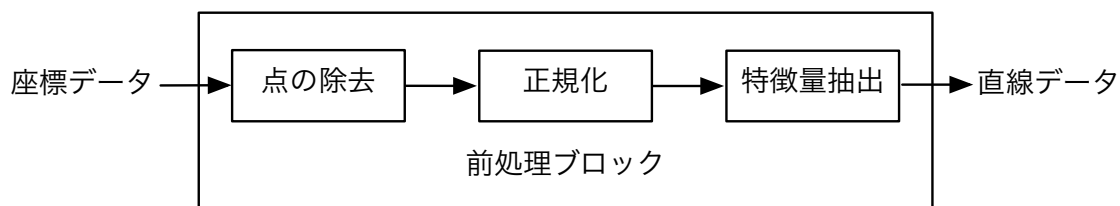


図 3.2: 前処理ブロックの概要

3.2 前処理ブロック

図 3.2 に前処理ブロックの概要を示す。前処理ブロックでは、点の除去、正規化、特徴量抽出を行う。直線上の点や近接する点といった、取り除いても文字として成り立つような点の除去を行い、その後正規化を行う。特徴量抽出では点データを直線データへ変換する。以下、前処理ブロックにおける各処理を示す。

3.2.1 余分な点の除去

図 3.3(a) にオリジナルの単語データの例と、図 3.3(b) に余分な点が除去された後の単語データの例を示す。収集する医療用語手書きデータは、図 2.2 で示したように (x, y) 座標の時系列データとして存在している。本論文では文献 [2] に従って、これに各点の筆順情報 s を合わせた (x, y, s) を収集する。1 つの単語を式 3.1 のように収集する。

$$[[x_1, y_1, s_1], [x_2, y_2, s_2], \dots, [x_n, y_n, s_n]] \quad (3.1)$$

x_i と y_i は点の座標、 s_i はその点が何画目のストローク上にあるかを示したものである。

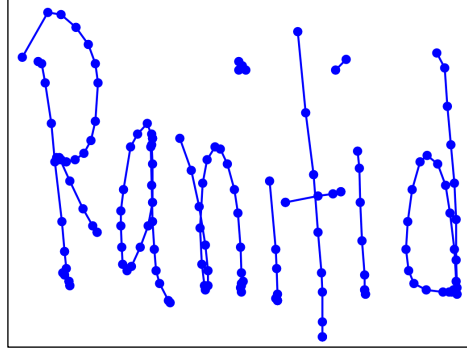
手書きデータは書くスピードの違いなどが原因で、同じ単語でもデータ提供者によって点の数が大きく異なってしまい、うまく認識ができない可能性がある。本論文では先行研究に従い、取り除いても文字として成り立つような点の除去を行うことでデータ提供者ごとの点の数の差を小さくする。

3.2.2 正規化

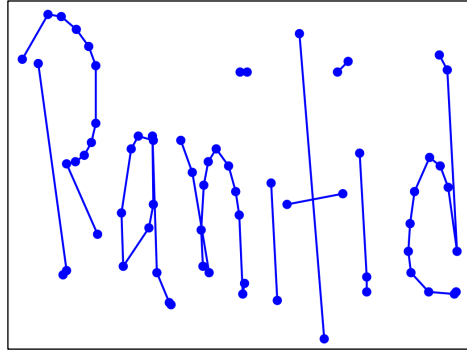
入力データをさらに簡潔にするため、点の除去を施したデータの正規化を行う。はじめに x 座標についての正規化の手順を説明する。全てのデータの中から最大値 x_{max} と最小値 x_{min} を取り出し、式 3.2 を用いてある点の x 座標 X を X_{nor} に正規化する。

$$X_{nor} = \frac{X - x_{min}}{x_{max} - x_{min}} \quad (3.2)$$

y 座標に対しても同様の計算を行い、結果的に (x, y) をそれぞれ 0 以上 1 以下のデータとする。



(a) オリジナルデータ



(b) 点の除去後

図 3.3: データ前処理：余分な点の除去

3.2.3 特徴量抽出

機械学習への入力のため、特徴量を抽出する．正規化した点データ 2 点間を繋ぎ、直線データを作成する．直線データに変換することで、点の座標だけでなく直線の長さ、角度などより多くの特徴量を抽出することができる．線データ L_i は、点 i と点 $i+1$ から式 3.3 のように構成され、本論文ではこのデータを学習における入力データとする．

$$L_i = [x_i, y_i, \Delta x_i, \Delta y_i, I(s_i = s_{i+1}), I(s_i \neq s_{i+1})] \quad (3.3)$$

ここで, $\Delta x_i = x_{i+1} - x_i$, $\Delta y_i = y_{i+1} - y_i$ であり, $I()$ は括弧内の条件が真であるときに 1 でありそれ以外では 0 である. L_i において x_i と y_i は直線の始点を表し, Δx_i と Δy_i は線の始点から終点までの x 軸方向, y 軸方向の距離を表す. また $I(s_i = s_{i+1}) = 1$ は, 直線の始点と終点が同一ストローク上にあることを示し, $I(s_i \neq s_{i+1}) = 1$ はその直線で次のストロークに移ったことを示す.

3.3 データ拡張ブロック

ここでは様々な手書き文字に対応するためのデータ拡張について説明する. 本研究では多様な提供者から大量のデータを得る必要があるが, 十分なデータ量を確保するためには非常に多くの労力と時間を要する. そのため複数のデータ拡張手法を適用することで機械学習の際の手書き文字データの多様性を高める. はじめに先行研究の SRP 手法を説明する. その後, 文字の縦横比を変更する変換する Ratio 手法を提案する.

3.3.1 SRP 手法

ここでは, 先行研究 [1] のデータ拡張手法である Stroke Rotation and Parallel-shift (SRP) 手法について説明する. SRP 手法とは, ストロークの回転と平行移動によってデータ量を水増しする手法である.

ストロークの回転

ストロークの始点と終点の座標から中点を求め, その点を中心にストローク上の点をそれぞれ回転させることでストローク全体を回転させる. この処理を 1 つのデータに対して複数回施すことで, データ拡張を行う.

図 3.4(a) にストロークの回転の原理を示す．ストロークの始点を (x_f, y_f) ，終点を (x_l, y_l) とする．始点と終点の中点を (a, b) とすると，式 3.4 より (a, b) の値が求められる．

$$(a, b) = \left(\frac{x_f + x_l}{2}, \frac{y_f + y_l}{2} \right) \quad (3.4)$$

ストローク上の任意の点の座標を (x, y) としたとき，その点を，点 (a, b) を中心に角 θ だけ移動させた後の座標 (X, Y) は 式 3.5 で表される．

$$\begin{pmatrix} X - a \\ Y - b \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x - a \\ y - b \end{pmatrix} \quad (3.5)$$

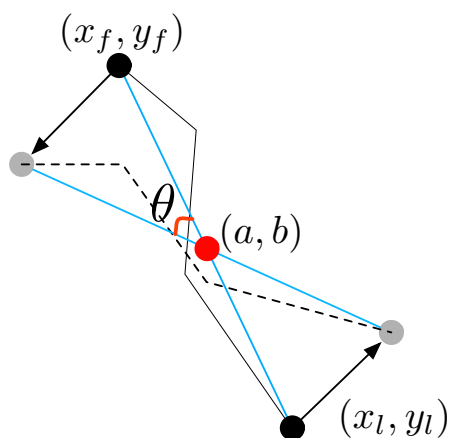
この式をストローク上のすべての点に用いることで，ストローク自体を角 θ 回転させる．図 3.4(b) にストローク回転前の単語データの例，図 3.4(c) にストローク回転後の単語データの例を示す．この処理を，ストロークごとに角度を変えながら行うことで元のデータとは異なる形の文字・単語を生成する．それを 1 つの単語データに対して N 回行い，データ量を N 倍に拡張する．

ストロークの平行移動

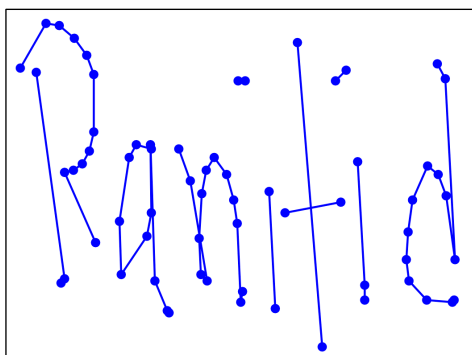
ストローク上の点の座標それぞれに一定の値を加え，ストローク全体を平行移動させることでデータ拡張を行う．図 3.5(a) にストロークの平行移動の原理を示す．ストローク上の任意の点の座標を (x, y) としたとき，その点を x 方向に dx ， y 方向に dy だけ平行移動させた後の座標 (X, Y) は 式 3.6 で表される．

$$(X, Y) = (x + dx, y + dy) \quad (3.6)$$

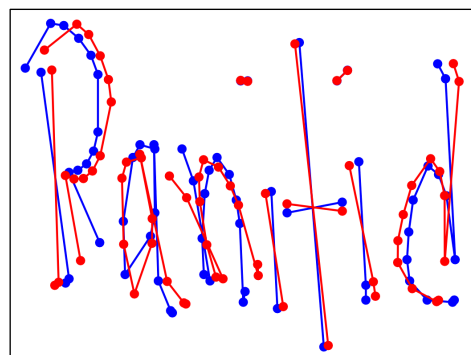
この式をストローク上のすべての点に用いることで，ストローク自体を x 方向に dx ， y 方向に dy 平行移動させる．図 3.5(b) にストローク平行移動前の単語データの例，



(a) 回転の原理



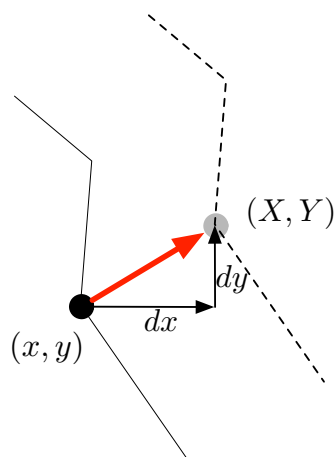
(b) データ前処理後



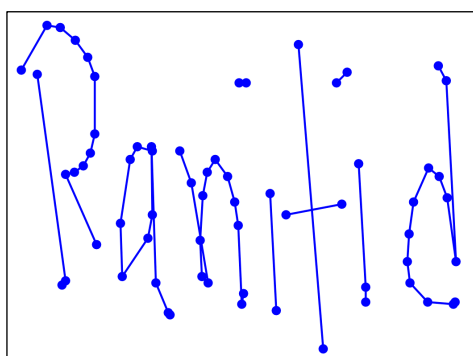
(c) 回転前（青）と回転後（赤）

図 3.4: ストロークの回転

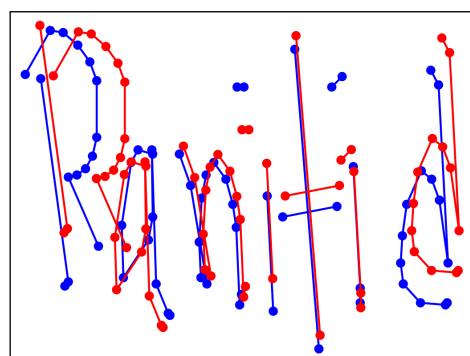
図 3.5(c) にストローク平行移動後の単語データの例を示す。この処理を，ストロークごとに dx と dy の値を変えながら行うことで元のデータとは異なる形の文字・単語を生成する。それを 1 つの単語データに対して N 回行い，データ量を N 倍に拡張する。



(a) 平行移動の原理



(b) データ前処理後



(c) 平行移動前（青）と平行移動後（赤）

図 3.5: ストロークの平行移動

3.3.2 Ratio 手法

ここでは、オンライン文字におけるデータ拡張の新技术として、文字の縦横比を変更する Ratio 手法を提案する。先行研究 [1] の SRP 手法において回転の大きさ、平行移動の大きさは筆者が文字の形を崩さないと判断できる範囲で設定している。そのためデータを 100 倍に拡張した場合、同じようなデータが増えてしまい学習精度が低くなる原因になる。Ratio 手法では文字の縦横比変更を行うので文字のストロークが重なることが

なく、文字の形が大きく崩れてしまうことがない。そのため文字の形を崩さない最大のパラメーターで SRP 手法を適用した後も、Ratio 手法を用いてさらに拡張できる。また Ratio 手法は事前に変更比率を定める手法であり、Ratio 手法を用いて拡張されたデータはそれぞれが必ず一定の割合で変化しており、少ない倍率でも Ratio 手法を組み合わせることによってデータが多様化する。SRP 手法と Ratio 手法を組み合わせることにより SRP 手法のみで拡張をする場合に比べ、よりデータが多様化を実現することができる。

図 3.6(a) に文字の縦横比変更の原理を示す。文字の縦横比変更は x 座標を基準に行う場合と y 座標を基準に行う場合がある。ここでは y 座標を基準に行う場合を具体的に説明する。1 つの単語におけるすべての点の y 座標の平均値を取り、すべての点の y 座標に対し平均値との大小関係の比較をする。比較結果に応じて、点の移動を行う。文字の拡張比率を r 、1 つの単語上におけるすべての点の y 座標平均を \bar{y} とし、同単語上の任意の点の座標を (x, y) としたとき、その点の y 座標が \bar{y} より大きい場合、 y 座標に y 座標の値の r 倍を加算し、 \bar{y} との距離が大きくなる方向に移動させた後の座標 (X, Y) は式 3.7 で表される。

$$(X, Y) = (x, y(1 + r)) \quad (3.7)$$

一方、 y 座標が \bar{y} より小さい場合、 y 座標から y 座標の値の r 倍を減算し、 \bar{y} との距離が大きくなる方向に移動させた後の座標 (X, Y) は式 3.8 で表される

$$(X, Y) = (x, y(1 - r)) \quad (3.8)$$

この式を単語上のすべての点に用いることで、単語全体の点を単語の y 座標の平均を基準に移動させ、単語全体を y 方向に引き伸ばす。 x 座標を基準に拡張する場合は、これらの式の x と y を逆にして計算を行う。

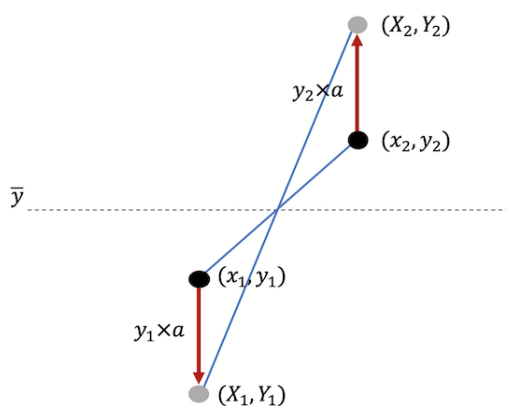
図 3.6(b) に縦横比率変更前の単語データの例，図 3.6(c) に文字の縦横比率を y 座標を基準に変更後の単語データの例を示す．この処理を，拡張回数ごとに拡張比率 r と，単語の x, y どちらを基準に適応するか変えながら行うことで元のデータとは異なる形の文字・単語を生成する．それを 1 つの単語データに対して N 回行い，データ量を N 倍に拡張する．

x 座標・ y 座標どちらを基準に比率変更を行うかは $x:y$ を 1:2 の割合でランダムに決定している．1 つの単語において文字が横方向に並んでいる場合には，全体を y 座標を基準に縦方向に比率変更を行うのが効果的である．一方単語が 1 箇所にも固まっている場合や斜め方向に書かれている場合には，全体を x 座標を基準に横方向に比率変更を行うのも効果的であると考えられる．本研究において収集したデータは英語とバングラ語を含んでおり，さらに一般的な文字より雑な文字を扱うため文字の配置が横方向のものだけでなく，斜めや 1 箇所にも集中したものを含んでいる．このため $x:y$ を 1:2 の割合とすることでより効果的にデータを多様化させている．

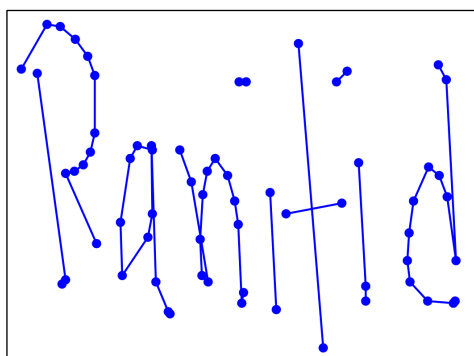
3.4 機械学習ブロック

本論文では BidirectionalLSTM を用いて学習を行う．図 3.7 に BidirectionalLSTM の概要を示す．BidirectionalLSTM は従来の LSTM に未来の入力から計算を行う逆方向のモデルを加え，出力を同一の出力層に統合するものである．学習プロセスでは，データ拡張が施された直線データを入力として用いる．本論文において BidirectionalLSTM は，現在入力されている直線データより前に書かれた直線データだけでなく，後に書かれる直線データも用いて学習を行う．推定プロセスでは，学習が行われたモデルに前処理後のデータを入力し，用語の推定を行う．

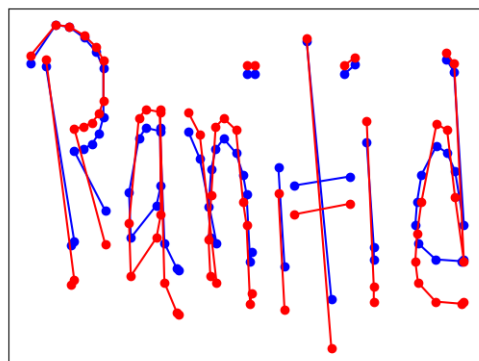
学習モデルは先行研究を参考に作成し，実装には python のニューラルネットワーク



(a) 縦横比率変更の原理



(b) データ前処理後



(c) 縦横比率変更前（青）と変更後（赤）

図 3.6: 文字の縦横比率変更

ライブラリである Keras [11] を用いている．活性化関数には Softmax 関数 [12] を用いて出力を確率として解釈し，最も確率の高い項目を予測値として決定している．損失関数には categorical_crossentropy [13]，最適化関数には Adam [14] を用いており，学習の評価は正解率で計算を行なっている．

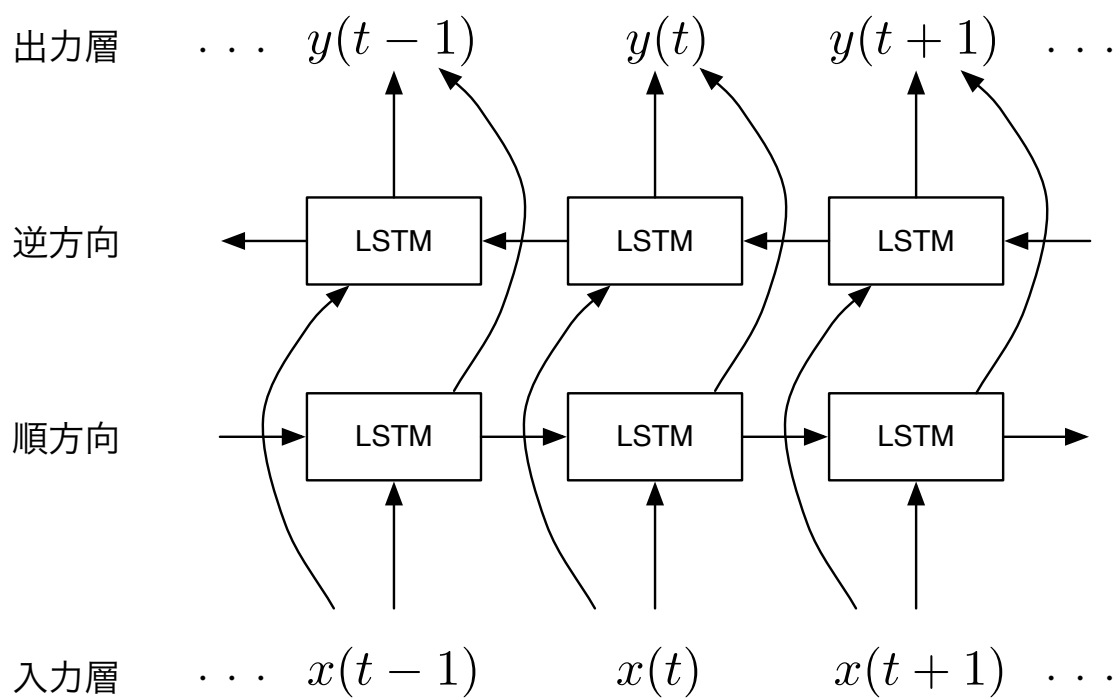


図 3.7: BidirectionalLSTM の概要

第4章

実装

本章では先行研究で収集したデータについて説明し，本論文でのオンライン手書き医療用語認識手法の実装について説明する．

4.1 データ収集

学習に使用したいデータがオープンソースで存在していないため，先行研究 [1] において独自に収集しているデータを用いる．本研究では，過去の処方箋データから医療用語コーパスを作成している．コーパスとは，言語を分析するための基礎資料として書き言葉や話し言葉の資料を収集し，研究用の情報を付与したものである．本研究及び先行研究では医療用語の手書き認識を行う．そこで 1.1 節で述べた PHC の過去の処方箋データからコーパスを作成し，それを機械学習の正解データとして用いている．8324 名分の過去の処方箋データは，それぞれの欄が症状，薬名などに分けられる．先行研究では薬名欄に頻出する単語（360 語，英語）と医者からのアドバイス欄に頻出する単語（120 語，バングラ語）を用いてコーパスを作成し，データ収集を行っている．データ提供者により手書きされた文字は正解データとともにデータベースに保存される．

表 4.1: 実装環境

OS	Ubuntu 16.04
GPU	NVIDIA GeForce 1080
メモリ	8GB
プロセッサ	3.70GHz Intel Core i7-8000K

4.2 使用機器・学習モデル構造

図 4.1 に使用機器，表 4.1 に実装環境を示す．機械学習には NVIDIA 社の GeForce 1080 が 1 枚搭載された デスクトップ PC を用いた．図 4.2 に本研究で用いた学習モデルの構造を示す．本研究で用いるデータはデータ長が一定ではないため，データの後ろを 0 でパディングすることでデータ長最大の値である 260 に揃え，学習の入力とした．LSTM 層の出力次元数は 300 に設定し，プールサイズの値の平均値をそれぞれ出力するプーリング層を配置した．その後パラメータの数を増やすために全結合層を配置した．なお，過学習を防ぐためプーリング層と全結合層の間にドロップアウト [15] をそれぞれ 0.3 の割合で設定した．



図 4.1: 使用機器

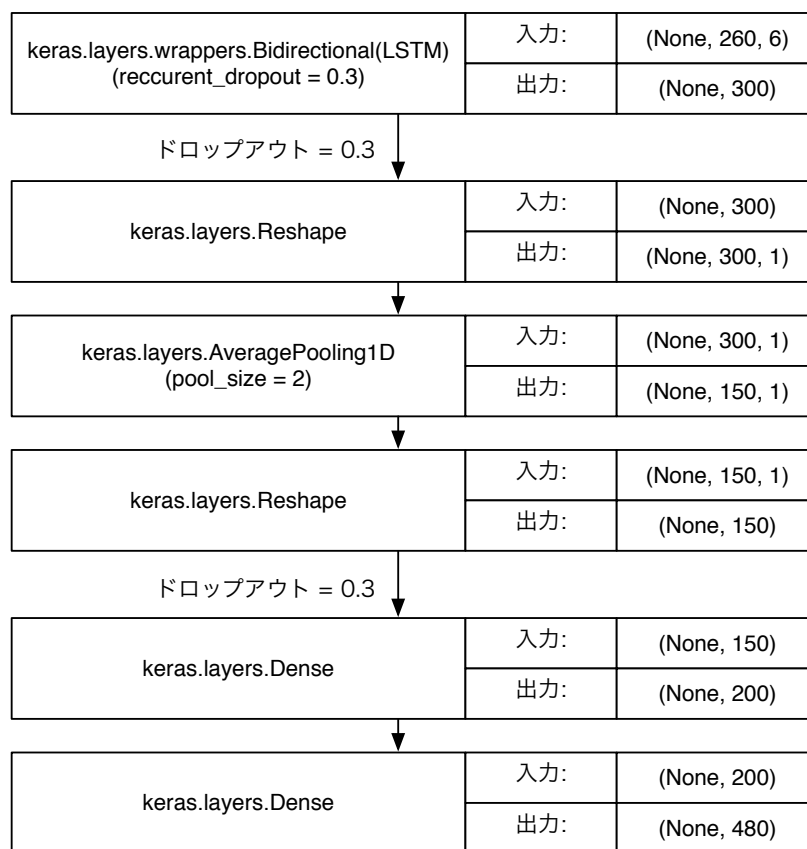


図 4.2: 層の構成

第5章

評価

本章では，4.1 節で 39 名から収集した 15911 語を用いて行った評価の概要と，その結果を示す．

5.1 各値の設定

表 5.1 に SRP 手法におけるデータ拡張時の各値を示し，表 5.2 に Ratio 手法におけるデータ拡張時の各値を示す．また表 5.3 にデータ拡張の種類・拡張倍率の組み合わせパターンを示す．データ拡張を使用したときは拡張後のデータ数を元データの 100 倍に統一している．回転でのデータ拡張では，ガウス分布を用いて平均 $\mu = 0$ ，分散 $\sigma^2 = 1.5$ となるような乱数の角度をストロークごとに生成し，ストロークの回転を施した．平行移動でのデータ拡張では，ガウス分布を用いて平均 $\mu = 0$ ，分散 $\sigma^2 = 0.0001$ となるような乱数の dx と dy を各ストロークごとに生成し，ストロークの平行移動を施した．文字の比率変更のデータ拡張では， x 座標・ y 座標どちらを基準に比率変更を行うかを 3.3.2 項をもとに， $x : y = 1 : 2$ の割合でランダムに決定している．また比率の割合を拡張回数が 1 回の時に $r = 0.00$ とし回数を重ねるごとに $r = 0.02$ ずつ増加させて，文字の拡

表 5.1: SRP 手法におけるデータ拡張時の値

データ拡張の種類	μ	σ^2
ストロークの回転移動	0	1.5
ストロークの平行移動	0	1.0×10^{-4}

表 5.2: Ratio 手法におけるデータ拡張時の値

データ拡張倍数	$x : y$ の割合	拡張比率 (拡張回数 n)
10 倍以下	1 : 2	$0.02 \times (n - 1)$
100 倍	1 : 2	$0.02 \times (n - 1)$

表 5.3: データ拡張の種類・拡張倍率の組み合わせ

平行移動 (PS) の倍率	回転 (SR) の倍率	文字の比率拡張の倍率
100.0	不使用	不使用
不使用	100.0	不使用
不使用	不使用	100.0
10.0	10.0	不使用
10.0	不使用	10.0
不使用	10.0	10.0
5.0	5.0	4.0

張比率の変更を施している．ただし拡張回数が 100 回の時は，比率の割合を $r = 0.002$ ずつ増加させている．先行研究ではストロークの平行移動，回転をそれぞれ 10 回ずつ行っている．本論文では，ストロークの平行移動，ストロークの回転，文字の比率変更を拡張回数を変更し，それぞれを組み合わせ実験を行なっている．

5.2 評価方法

データ提供者 39 名の中から、データ提供が途中で止まっていた 12 名を除いた 27 名のうちから 3 名をランダムで抜き出し、その 3 名のデータをテスト用のデータとし、残りの 24 名と 12 名のデータを学習用のデータとした。その後学習用データに対してデータ拡張を行い、検証用データと 9 : 1 の割合で分けた後機械学習を行った。学習用データにはテストに用いられるデータ提供者 3 名のデータは一語も含まれていない。この過程を 10 回繰り返し、各回の精度の平均値、中央値などを求める。以上の方法で表 5.3 に示したパターンでのそれぞれの精度を求め、先行研究 [1] の結果と比較することで本研究のデータ拡張手法の有効性を測る。

5.3 評価結果

表 5.4 に先行研究および本研究での評価結果を示す。データ拡張を行った際はエポック数を 5、ミニバッチサイズを 512 に設定し、データ拡張を行わなかった場合はエポック数を 100、ミニバッチサイズを 64 に設定した。エポック毎に異なるミニバッチを作成するため、エポック毎に学習データをシャッフルした。過学習を防ぐため、どちらにも keras の EarlyStopping [16] を用いた。最適化関数には Adam [14] を用い、学習率は 0.001 とした。

結果として、データ拡張を行わなかった場合の平均精度が 73.4%、SRP 手法のみでデータを 100 倍に拡張した場合の平均精度は 89.5%であったのに対し、SRP 手法と新手法 Ratio 手法を組み合わせで拡張した場合の平均精度は 93.0%であった。先行研究 [1] において、データ拡張を行わなかった場合、SRP 手法のみでデータ拡張を行った場合でも最大精度は比較的高いが、最小精度が低かった。

ストロークの回転手法（SR 手法）のみを用いてデータを 100 倍にした場合では、最

小精度が大きく減少した。これは、回転手法でのデータ拡張を 100 回も行うと同じようなデータが増えてしまい、過学習が起きてしまうことが原因であると考えられる。データ拡張におけるパラメータは文字の形が変わらないように設定している。文字においてそれぞれのストロークの角度は、ストロークの長さや位置よりも重要な要素であり、回転手法において文字の形を変えないようにするためには回転拡張時のパラメータがとる値の範囲を狭める必要がある。そのため回転手法のみでデータ拡張を行うと、拡張の度にランダムに設定されたパラメータの値によっては似たようなデータが増えてしまい過学習を起こしてしまう。ストロークの平行移動（PS 手法）、文字の比率変更（Ratio 手法）では文字のストロークと点の平行移動を行っており拡張方法が類似しているため、同じような精度を得た。最小精度が高い要因として、どちらの手法も拡張後に文字の形を大きく変えてしまうことがなく、文字の形を残したままデータを多様化することができたからである。

それぞれの拡張手法を組み合わせる拡張を行った場合、拡張倍数が 100 倍の場合ではいずれの場合でも最小精度が低かった。先行研究 [1] の SRP 手法でデータ拡張を行った場合には、回転手法で同じようなデータを増やしてしまい、平行移動による手法のみの場合に比べてデータが多様化されなかったことが、最小精度を低くした原因だと考えられる。新手法の Ratio 手法を用いた場合では、SRP 手法に比べて最小精度を上げることができている。これは Ratio 手法が事前に変更比率を定める手法であり、Ratio 手法を用いて拡張されたデータはそれぞれが必ず一定の割合で変化しており、少ない倍率でも Ratio 手法を組み合わせることによってデータが多様化するからである。しかし PS 手法と Ratio 手法を組み合わせただけの場合では、それぞれを単独で 100 倍に拡張した場合より、平均精度および最小精度が低くなっている。これは、PS 手法と Ratio 手法が似た手法であり Ratio 手法で拡張されたデータに PS 手法のみで拡張を施すと、同じようなデータが増えてしまったことが原因だと考えられる。

表 5.4: 評価結果

評価対象	拡張倍数	平均精度 [%]	最大精度 [%]	最小精度 [%]
データ拡張なし	1.0 倍	73.4	90.8	19.0
PS(100 回)	100.0 倍	91.3	93.9	87.8
SR(100 回)	100.0 倍	83.8	95.6	3.33
Ratio(100 回)	100.0 倍	91.4	95.7	87.5
(従来手法) SR(10 回)PS(10 回)	100.0 倍	89.5	95.4	68.0
SR(10 回)Ratio(10 回)	100.0 倍	90.0	95.9	77.4
PS(10 回)Ratio(10 回)	100.0 倍	88.8	94.2	78.0
SR(5 回)PS(5 回)Ratio(4 回)	100.0 倍	93.0	94.5	92.1

SR 手法及び PS 手法を 5 回ずつ用いて、Ratio 手法を 4 回適用し 100 倍に拡張した場合では、平均精度及び最小精度を大きく向上させることができています。SR 手法を 100 倍で拡張した際は SR 手法のパラメータの範囲が狭いために同じようなパラメータを選択してしまっていたが、5 倍の拡張では同じパラメータを選択する機会が少なかったことで、PS 手法及び Ratio 手法で拡張したデータをさらに多様化することができたからである。

上記結果より Ratio 手法を組み合わせることでデータ拡張を行った場合、Ratio 手法での拡張回数が少ない場合でもデータの多様化を行うことができるため、Ratio 手法を取り入れない場合の拡張方法に比べて最小精度が向上した。Ratio 手法はデータの多様化に対する有効性があるといえる。

第6章

おわりに

6.1 本研究の主たる成果

本論文では、遠隔医療における処方箋予測に向けたオンライン手書き医療用語認識における、オンライン文字認識のデータ拡張手法である Ratio 手法の提案を行った。まず最初に先行研究でのデータ拡張では、拡張後に同じようなデータが増えてしまい最小精度が低くなってしまうと指摘した。この問題を解決するために Ratio 手法の提案と SRP 手法と組み合わせた際の精度を安定させる拡張倍率の分析を行った。新手法 Ratio を組み合わせたデータ拡張手法の精度評価の結果、収集した 15991 語のデータと 480 語のクラスにおいて平均精度 93.0%，最小精度 92.1%で単語を認識した。この結果は SRP 手法のみの場合の認識精度と比べて平均精度は 3.5%，最小精度は 14.7%向上した。データ拡張を行わなかった場合の認識精度と比べて平均精度は 19.6%，最小精度は 73.1%向上した。

6.2 今後の課題

今後の課題としては以下の 3 つがあげられる。1 つ目は拡張手法の拡張倍率とパラメータの調整である。機械学習において似たようなデータが増えた状態で学習を行うと、学習データに対する過学習で精度が低くなってしまう。また機械学習や前処理で多くのデータを利用すると、コンピュータへの負荷や学習時間が増加してしまうため学習に良い影響を及ぼさないデータは余計に増やすべきではない。そのため、似たようなデータが増えない最小の拡張倍数・パラメータでデータ拡張を行い精度を向上させる必要がある。本研究ではデータを 100 倍に拡張してデータ拡張を行ったが、それぞれの拡張倍数の組み合わせや拡張時のパラメータは実験を重ねて考察する必要がある。

2 つ目は提案したデータ拡張手法の応用である。本研究では独自に用意したデータを用いたが、提案した手法をオープンセットのデータセットに適用することで提案手法の汎用性を考察する必要がある。機械学習においてデータ量は精度を大きく左右する重要なパラメータである。本研究でのデータ拡張が他にどのような場面で応用できるか、検討を続ける必要がある。

3 つ目は本論文で行ったオンライン手書き医療用語認識を利用した処方箋予測である。本論文において単語の関係性などは考慮していないが、処方箋予測を行う際には単語同士を関連づける必要がある。1 つのメモ内における単語の関係性を学習の際の特徴量に加えることで、より高精度の文字認識が実現でき正確な処方箋予測につながると考えられる。単語同士の関係性や医者メモの分析などは、今後処方箋予測を行う上で考察を行う必要がある。

謝辞

本研究の機会を与えてくださり，様々なご指導をいただきました九州大学大学院システム情報科学研究院のアシル准教授に深く感謝いたします。本研究において，有益なご助言を頂きました九州大学大学院システム情報科学研究院の福田晃教授，九州大学 LSI 研究センターの久住憲嗣准教授，九州大学大学院システム情報科学研究院の石田繁巳助教，本研究室の高橋遼さんに深く感謝いたします。最後に，日頃の研究活動において様々な協力を頂きました，九州大学大学院システム情報科学研究院福田・久住・アシル研究室諸氏に深く感謝し，御礼を申し上げます。

参考文献

- [1] 高橋遼, 遠隔医療システムにおける処方箋予測に向けた手書き医療用語認識に関する研究, 九州大学, 2019.
- [2] X.-Y. Zhang, F. Yin, Y.-M. Zhang, C.-L. Liu, and Y. Bengio, “Drawing and recognizing chinese characters with recurrent neural network,” *IEEE transactions on pattern analysis and machine intelligence*, vol.40, no.4, pp.849–862, 2018.
- [3] W. Yang, L. Jin, and M. Liu, “Chinese character-level writer identification using path signature feature, Drop Stroke and deep CNN,” *Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR)*, pp.546–550, 2015.
- [4] A. Ahmed, A. Rebeiro-Hargrave, Y. Nohara, R.I. Maruf, P.P. Ghosh, N. Nakashima, and H. Yasuura, “Portable health clinic: A telehealthcare system for unreached communities,” *Smart sensors and systems*, pp.447–467, Springer, 2015.
- [5] A. Yuan, G. Bai, L. Jiao, and Y. Liu, “Offline handwritten english character recognition based on convolutional neural network,” *Proceedings of the IEEE 10th*

- IAPR International Workshop on Document Analysis Systems (DAS), pp.125–129,, 2012.
- [6] FKI, “IAM On-Line Handwriting Database”. <http://www.fki.inf.unibe.ch/databases/iam-on-line-handwriting-database>.
- [7] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, “CASIA online and offline Chinese handwriting databases,” Proceedings of the IEEE 12th International Conference on Document Analysis and Recognition (ICDAR), pp.37–41,, 2011.
- [8] B. Chakraborty, P.S. Mukherjee, and U. Bhattacharya, “Bangla online handwriting recognition using recurrent neural network architecture,” Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing ACM, pp.63, 2016.
- [9] 岡谷貴之, 深層学習, 講談社, 2015.
- [10] W. Yang, L. Jin, and M. Liu, “Deepwriterid: An end-to-end online text-independent writer identification system,” IEEE Intelligent Systems, vol.31, no.2, pp.45–53, 2016.
- [11] “Keras”. <https://keras.io/ja/>.
- [12] keras, “Softmax 関数”. <https://keras.io/ja/activations/>.
- [13] keras, “categorical_crossentropy”. <https://keras.io/ja/objectives/>.
- [14] D.P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014.
- [15] keras, “Core レイヤー”. <https://keras.io/ja/layers/core/>.

- [16] keras, “コールバック”. <https://keras.io/ja/callbacks/>.