テキスト分析における Character-level CNN の性能評価 - NTCIR-13 MedWeb タスクを題材として -

○宮崎和光 井田正明 (独立行政法人 大学改革支援・学位授与機構)

Evaluation of Character-level CNN in Text Analysis: Case Study in NTCIR-13 MedWeb Task

∗K. Miyazaki and M. Ida (National Institution for Academic Degrees and Quality Enhancement of Higher Education)

Abstract– テキスト分析手法として Character-level CNN (CLCNN) が注目されている. 著者らはこれまでに, 大学における 3 つのポリシーの分析に CLCNN を利用し, その有効性を確認してきた. しかしそこでは, マルチクラス問題への適用に留まっており, ひとつのサンプルが複数のクラスに属する可能性のあるマルチラベル問題での有効性は未検証のままであった. そこで本論文では, マルチラベル問題のテストコレクションとして知られる NTCIR-13 MedWeb タスクを題材に, CLCNN の有効性を実験的に検証する. 特に, ユニット数の違いによる性能変化や, Grad-CAM を用いた学習結果の可視化に注目した議論を行う.

Key Words: テキスト分析, 深層学習, Character-level CNN, NTCIR-13 MedWeb タスク, 学習結果の可視化, Grad-CAM

1 はじめに

医療データでは膨大な量の電子データを扱う必要がある. 画像データに対しては, 深層学習を利用した診断サポート [3] など, 多くの成果がある. 一方, 医療分野では画像とともに多くのテキストデータが存在する. 特に, 患者の生の声からの医療診断が実現されることが期待される. そのような患者の生の声の模擬例として, NTCIR-13 MedWeb (Medical Natural Language Processing for Web Document) [22] が知られている. そこで本稿では, NTCIR-13 MedWeb で公開されているデータを題材に, 深層学習の医療診断への適用可能性を検討する.

本稿では, テキストデータの分類を扱う. テキストデータを分類する方法として, Character-level CNN (CLCNN) [25] が知られている. 深層学習は, 一般に, 画像分類に効果的であるが, CLCNN は, その効果をテキスト分類へ拡張したものである.

著者らは, これまでに, 大学における卒業認定・学位授与の方針であるディプロマ・ポリシーが, 授与される学位に付記する専攻分野の名称を適切に表象しているかを検証するためのマッチングテストに CLCNN を適用し, その有効性を確認している [12, 15]. また, ディプロマ・ポリシーとカリキュラム・ポリシーの整合性においても, CLCNN の有効性を確認している [13].

しかしこれらの適用例においては, マルチクラス問題への適用に留まっており, ひとつのサンプルが複数のクラスに属する可能性のある**マルチラベル問題**での有効性は未検証のままであった. そこで本論文では, マルチラベル問題のテストコレクションとして知られる NTCIR-13 MedWeb タスクを題材に CLCNN の性能評価を行う.

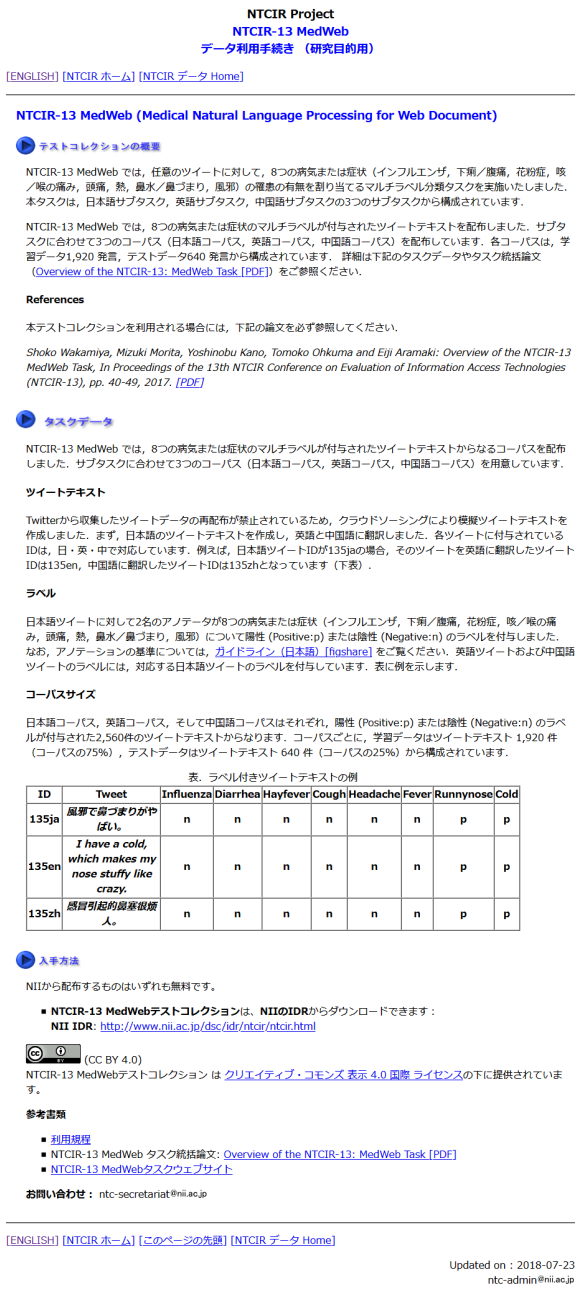
CLCNN を NTCIR-13 MedWeb タスクに適用した例として文献 [5] がある. しかし, そこでは複数の学習器を組み合わせたアンサンブル学習での性能向上がメインであり, CLCNN 単独での性能が詳しく評価されているわけではない. そこで, 本稿では, マルチラベル問題での CLCNN 単独の性能を評価するために, NTCIR-13 MedWeb タスクを用いた各種の実験を行う. 特に, 文献 [1] で紹介されているユニット数の違いによる性能変化や, Grad-CAM を用いた学習結果の可視化に注目した議論を行う.

2 NTCIR-13 MedWeb Task

医療分野には, 画像とともに多くのテキストデータが存在する. 特に, 患者の生の声から医療診断が実現されることが期待されている. NTCIR-13 MedWeb タスクは, そのような患者の生の声の模擬例として知られている.

Fig. 1 に示した Web ページ [23] のスクリーンショットをもとに, NTCIR-13 MedWeb タスクについて説明する. NTCIR-13MedWeb タスクでは, 8 種類の「病気/症状」のラベルが割り当てられた疑似ツイートデータを扱う. 各疑似ツイートデータには, 8 種類の各々の「病気/症状」に対し, その「病気/症状」が「ある」ことを意味する Positive(p), その「病気/症状」が「ない」ことを意味する Negative(n) のいずれかのラベルが付与されている.

このタスクは, 3 つの言語 (日本語, 英語, 中国語) で提供されている. 「病気/症状」には, { インフルエンザ (Influenza), 下痢/腹痛 (Diarrhea), 花粉症 (Hayfever), 咳/喉の痛み (Cough), 頭痛 (Headache), 熱 (Fever), 鼻水/鼻づまり (Runnynose), 風邪 (Cold)} の 8 種類を想定している. 日本語, 英語, 中国語のコーパスは, それぞれ 2,560 のツイートデータで構成されている. 各コーパスは, 1,920 のツイートデータ (コーパス全体の 75%) で構成される学習データと, 640 のツイートデータ (コーパス全体の 25%) で構成されるテストデータに分かれている. なお, 詳細については, 文献 [22] を参照されたい.



NTCIR Project
NTCIR-13 MedWeb
データ利用手続き（研究目的用）

[ENGLISH] [NTCIR ホーム] [NTCIR データ Home]

NTCIR-13 MedWeb (Medical Natural Language Processing for Web Document)

テストコレクションの概要

NTCIR-13 MedWeb では，任意のツイートに対して，8つの病気または症状（インフルエンザ，下痢／腹痛，花粉症，咳／喉の痛み，頭痛，熱，鼻水／鼻づまり，風邪）の罹患の有無を割り当てるマルチラベル分類タスクを実施いたしました．本タスクは，日本語サブタスク，英語サブタスク，中国語サブタスクの3つのサブタスクから構成されています．

NTCIR-13 MedWeb では，8つの病気または症状のマルチラベルが付与されたツイートテキストを配布しました．サブタスクに合わせて3つのコーパス（日本語コーパス，英語コーパス，中国語コーパス）を配布しています．各コーパスは，学習データ1,920 発言，テストデータ640 発言から構成されています． 詳細は下記のタスクデータやタスク統括論文（Overview of the NTCIR-13: MedWeb Task [PDF]）をご参照ください．

References

本テストコレクションを利用される場合には，下記の論文を必ず参照してください．

Shoko Wakamiya, Mizuki Morita, Yoshinobu Kano, Tomoko Ohkuma and Eiji Aramaki: Overview of the NTCIR-13 MedWeb Task, In Proceedings of the 13th NTCIR Conference on Evaluation of Information Access Technologies (NTCIR-13), pp. 40-49, 2017. [PDF]

タスクデータ

NTCIR-13 MedWeb では，8つの病気または症状のマルチラベルが付与されたツイートテキストからなるコーパスを配布しました．サブタスクに合わせて3つのコーパス（日本語コーパス，英語コーパス，中国語コーパス）を用意しています．

ツイートテキスト

Twitterから収集したツイートデータの再配布が禁止されているため，クラウドソーシングにより模擬ツイートテキストを作成しました．まず，日本語のツイートテキストを作成し，英語と中国語に翻訳しました．各ツイートに付与されているIDは，日・英・中で対応しています．例えば，日本語ツイートIDが135jaの場合，そのツイートを英語に翻訳したツイートIDは135en，中国語に翻訳したツイートIDは135zhとなっています（下表）．

ラベル

日本語ツイートに対して2名のアノテータが8つの病気または症状（インフルエンザ，下痢／腹痛，花粉症，咳／喉の痛み，頭痛，熱，鼻水／鼻づまり，風邪）について陽性 (Positive:p) または陰性 (Negative:n) のラベルを付与しました．なお，アノテーションの基準については，ガイドライン（日本語）[figshare] をご覧ください．英語ツイートおよび中国語ツイートのラベルには，対応する日本語ツイートのラベルを付与しています．表に例を示します．

コーパスサイズ

日本語コーパス，英語コーパス，そして中国語コーパスはそれぞれ，陽性 (Positive:p) または陰性 (Negative:n) のラベルが付与された2,560件のツイートテキストからなります．コーパスごとに，学習データはツイートテキスト 1,920 件（コーパスの75%），テストデータはツイートテキスト 640 件（コーパスの25%）から構成されています．

表．ラベル付きツイートテキストの例

ID	Tweet	Influenza	Diarrhea	Hayfever	Cough	Headache	Fever	Runnynose	Cold
135ja	*風邪で鼻づまりがやばい。*	n	n	n	n	n	n	p	p
135en	*I have a cold, which makes my nose stuffy like crazy.*	n	n	n	n	n	n	p	p
135zh	*感冒引起的鼻塞很烦人。*	n	n	n	n	n	n	p	p

入手方法

NIIから配布するものはいずれも無料です。

- **NTCIR-13 MedWebテストコレクション**は、**NIIのIDR**からダウンロードできます：
  NII IDR: http://www.nii.ac.jp/dsc/idr/ntcir/ntcir.html

(CC BY 4.0)
NTCIR-13 MedWebテストコレクション は クリエイティブ・コモンズ 表示 4.0 国際 ライセンスの下に提供されています。

参考書類

- 利用規程
- NTCIR-13 MedWeb タスク統括論文: Overview of the NTCIR-13: MedWeb Task [PDF]
- NTCIR-13 MedWebタスクウェブサイト

お問い合わせ： ntc-secretariat@nii.ac.jp

[ENGLISH] [NTCIR ホーム] [このページの先頭] [NTCIR データ Home]

Updated on：2018-07-23
ntc-admin@nii.ac.jp

Fig. 1: NTCIR-13 MedWeb タスクページ [23] のスクリーンショット

3 Character-level CNN を用いた NTCIR-13 MedWeb タスクの分類

3.1 基本方針

本稿が対象とする課題は，テキスト分類問題と捉えることができる．分類問題を解決する手法は，伝統的には，サポートベクターマシンなど様々な手法が提案されている．その中でも，近年，深層学習 (Deep Learning) が注目を集めており，本稿においても，深層学習による分類を試みる．

深層学習は画像分類 [9] やゲーム問題 [17, 19, 11] に威力を発揮している．そこでは，通常，畳み込みニューラルネットワーク (Convolutional Neural Network: CNN) を用いた学習が行われる．例えば，CNN に繰り返し，任意の画像と，それぞれの画像の意味 (正解) を教師信号として与えることで，未知の画像に対する分類が実現される．それに対し，本稿で扱うツイートデータなどの文書情報では，画像の代わりに自然言語を扱う必要がある．

3.2 Character-level CNN

自然言語処理技術は，近年，word2vec [10] や GliVe [20] などに代表される分散表現技術等の進展により著しい進歩を遂げている．一般に，深層学習を自然言語処理に応用する際には，ネットワークへの入力をどのように構成するかが問題となる． 文章を単語単位に分割し word2vec の出力値をネットワークの入力にする場合 [7] もあるが，本稿では，より汎用的な Character-level Convolutional Neural Network [25] (CLCNN) と呼ばれる技術を利用する．そこでは，文章を文字単位に分割した上で，各々を文字コード (例えば，unicode 値) に変換しネットワークへ入力する．その結果，文章をあたかも画像のように扱うことが可能となる．CLCNN は，例えば，口コミでおいしいお店を探せる Web サービス Retty [26] などで活用されている．

本稿で用いたネットワークの概要を Fig. 2 に示す．各文字をネットワークに入力する際には unicode 値を利用した．畳み込みは，同一の入力に対し，複数のカーネルサイズで実施した．ここで，カーネルサイズは，横は 1 文字の次元サイズ分，縦は 2, 3, 4 および 5 文字分の 4 種類を用いた．これにより，畳み込みの結果が n-gram を求めたようになることを期待している．複数のカーネルで畳み込みした結果をそれぞれプーリング層に流した後，64 個のユニットからなる全結合層にかけ，最終的にそれぞれが各「病気/症状」の有無に対応する 8 個の出力を得る．

なお，全結合層の出力に対して，Batch Normalization 処理をした後に，0.5 の確率で Dropout 処理 [21] を行っている．また，活性化関数としては，出力層ではシグモイド関数 (Sigmoid)，それ以外ではランプ関数 (ReLU) を用いた．

3.3 マルチクラス問題とマルチラベル問題

著者らは，これまでにも文献 [16, 14] において NTCIR-13 MedWeb タスクにおける CLCNN の有効性を検証している．しかし，そこでは，本来マルチラベル問題である NTCIR-13 MedWeb タスクを，各「病気/症状」ごとに，その対象とする「病気/症状」の有無の判定するというタスクに変換し解いていた．

マルチクラス問題では，一般に分類対象となるクラスが複数存在し，あるひとつのサンプルをどれかひとつのクラスに割り当てるような問題として定式化される．よく知られている MNIST や CIFAR などのデータセットがこれに相当する．それに対し，マルチラベル問題では，マルチクラス問題における「サンプルに対してクラスはひとつ」という制約を取り除いた問題として定式化される．そのため，一般に，マルチラベル問題はマルチクラス問題よりも解決が困難となる．

深層学習を用いてマルチラベル問題を解く場合，損失関数や出力層の活性化関数に工夫が必要となる．マルチクラス問題では，これらの関数に交差エントロピー (categorical_crossentropy) やソフトマックス関数 (softmax) を用いることが多い． それに対し，マルチラベル問題では，損失関数としては平均二乗誤

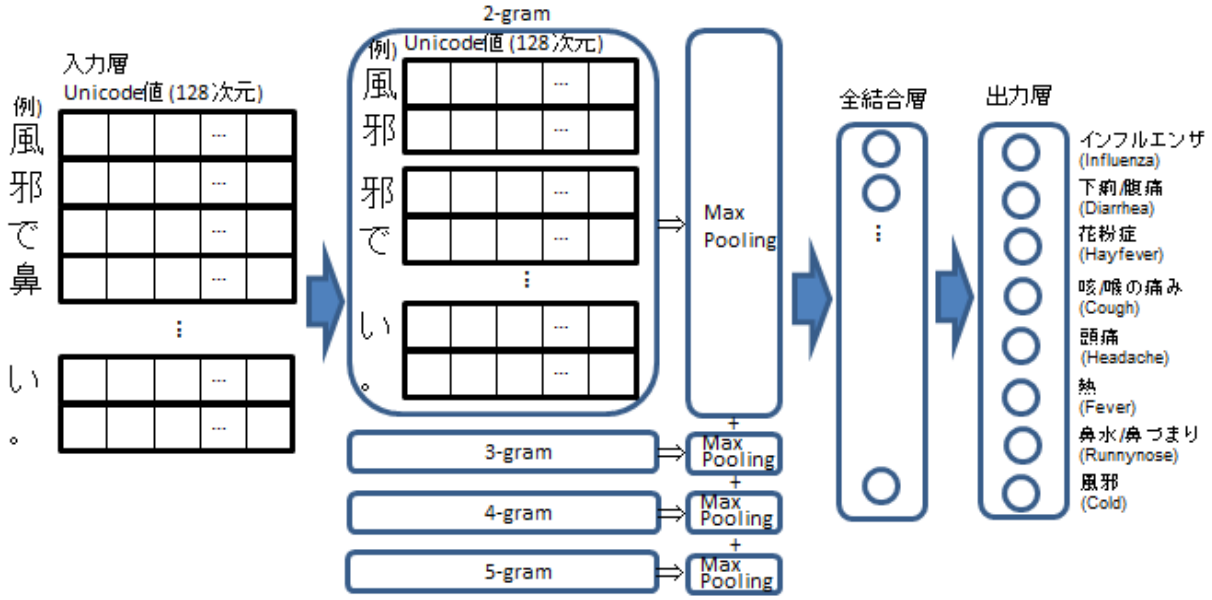


Fig. 2: 実験で用いたネットワーク構造の概要

差 (mean_squared_error) や二値交差エントロピー (binary_crossentropy), 出力層の活性化関数としてはシグモイド関数が一般的に用いられる. そこで, 本論文でも損失関数には平均二乗誤差, 出力層の活性化関数にはシグモイド関数を用いた.

4 NTCIR-13 MedWeb タスクにおける Character-level CNN の有効性の検証

4.1 検証方法

日本語版の NTCIR-13 MedWeb タスクに CLCNN を適用した. 各ツイートデータに対し, 各「病気/症状」への帰属度の学習を行う. 帰属度は各病気/症状ごとに, 0.0〜1.0 の値を取り, 1.0 に近い程, 入力したツイートデータに対し, その各「病気/症状」である可能性が高いことを意味するようにネットワークを学習させる.

全体的な処理の流れは以下のようになる.

1. 学習データの各単語を 128 次元で表される Unicode 値に変換する.
2. 手順 1 で取得した値を Fig. 2 のネットワークに入力する.
3. 入力したツイートデータが各「病気/症状」である場合, その「病気/症状」の出力値が 1.0, それ以外の「病気/症状」の出力値が 0.0 になるようにネットワークパラメータを更新する. なお, 本タスクはマルチラベル問題のため, ひとつのツイートデータに対し, 1.0 が期待される「病気/症状」が複数存在する可能性がある点には注意されたい.

結果を検証する際には, 学習済みのネットワークにテストデータを入力する. 例えば, 「風邪で鼻づまりがやばい」というツイートが入力された場合には, 「鼻水/鼻づまり」と「風邪」に対する出力が 1.0 に近く, それ以外の「病気/症状」に対する出力が 0.0 に近い場合, 学習は成功したと見なされる. そうでなければ, 学習が失敗したと判断される.

4.2 予備実験の結果

本稿では, ネットワークの出力値が 0.0〜1.0 の範囲になるように学習している. そこで, ネットワークの出力値が 0.5 以上のとき, その出力の「病気/症状」が「ある」と判断し, 0.5 未満のとき, その出力の「病気/症状」が「ない」と判断した上で、8 種類の出力の値が正解と完全に一致したときのみ 1.0, それ以外, すなわち, ひとつでも「病気/症状」の有無が一致していなければ 0.0 とする**完全一致率** (Exact match) という基準で評価することにした.

この基準で乱数の種を変えて 30 回実験を行ったときの, テストデータに対する完全一致率は, 平均 0.835 (標準偏差 0.007) であった. なお, 本稿で行った実験では, 多くの場合で, 「病気/症状」が「ある」場合の出力値は 0.9 以上, 「病気/症状」が「ない」場合の出力値は 0.1 よりも小さい値になるので, 「病気/症状」の有無を判断する閾値である 0.5 をより厳しい値に設定しても結果に大きな違いは生じなかった.

4.3 ユニット数の違いによる性能比較

Table 1: 全結合層のユニット数を変えたときの学習データに対する完全一致率の平均と標準偏差

	64	1000	10000	100000
ave.	0.973	0.980	0.957	0.941
S.D.	0.038	0.006	0.128	0.128

Table 2: 全結合層のユニット数を変えたときのテストデータに対する完全一致率の平均と標準偏差

	64	1000	10000	100000
ave.	0.836	0.835	0.840	0.837
S.D.	0.007	0.009	0.010	0.008

文献 [1] では, パラメータ数をデータ数よりも十分に大きくすることで, 極小解に捉われない解が得られるという Kawaguchi らの定理 [6] が紹介されている. そこ

で, ここでは, 全結合層のユニット数を 64, 1000, 10000, 100000 と増加させ, Kawaguchi らの定理に従った結果が得られるかを確認した. Table 1 および Table 2 に, 学習データおよびテストデータに対する完全一致率の平均 (ave.) と標準偏差 (S.D.) を示す. このふたつの表において, それぞれユニット数の違いによる有意差があるかを 2 標本 t 検定で検証したところ, 有意水準 1%で, いずれに値の間にも「有意差なし」となった.

Table 3: Dropout 停止時における全結合層のユニット数を変えたときの学習データに対する結果

	64	1000	10000	100000
ave.	0.889	0.778	0.979	0.963
S.D.	0.053	0.032	0.002	0.003

Table 4: Dropout 停止時における全結合層のユニット数を変えたときのテストデータに対する結果

	64	1000	10000	100000
ave.	0.772	0.714	0.835	0.834
S.D.	0.037	0.026	0.009	0.007

この結果は, Kawaguchi らの定理と異なるが, いずれのデータも十分高い性能を得ているため差が生じなかったものと考える. そこで, 深層学習において過剰適合を避ける方法としてよく利用されている Dropout [21] の停止を考える. Dropout は, 学習時に状態変数の一部をランダムに 0 に設定することによって, 特定の状態変数だけを重視した学習を防ぐ方法である. 3.2 節で述べたようにこの処理を本稿では, 全結合層の出力に対し 0.5 の確率で適用していた. Table 3 および Table 4 が, それぞれ, Dropout 停止時における, 学習データおよびテストデータに対する完全一致率の平均 (ave.) と標準偏差 (S.D.) である. これらの結果に対し, 先ほど同様検定したところ, テストデータに対する 10000 と 100000 の間以外には, 有意水準 1%で有意差あり, という結果が得られた. これは Kawaguchi らの定理と一致する. この結果からも, Dropout は, 極小解への有効な対策方法であることがあらためて確認できた.

さらに, 興味深いことに, Table 4 では, ユニット数が 1000 個のときに一時的に性能が悪化している. なお, 既に述べたようにこの悪化にも有意水準 1%で有意差あり, という結果が得られている. 文献 [1] では, Kawaguchi らの定理とともに, パラメータ数 P とデータ数 N の関係について, $P > N$ のときのテストデータに対する誤差 (**汎化誤差**) に関する理論 [2, 4] が同時に紹介されている. そこでは, 「単純な線形回帰問題で, 説明変数の数 P を大きくすると, 一度下がった汎化誤差は P を大きくしていくとあるところから上がってしまう. しかし, さらに N 以上に大きくしていくとまた下がるという理論がある.」(文献 [1] p.186) と記述されている. 今回実験で用いたデータは, 学習データ 1,920 個, テストデータ 640 個なので, 全結合層のユニット数が 1000 個のところで汎化誤差の変化が生じていることは, これらの理論 [2, 4] が示唆することと一致する. 回帰問題ではなく分類問題である NTCIR-13 MedWeb タスクにおいて, 回帰問題に対し構築された理論に合った結果が生じたことはたいへん興味深く, 今後の理論の発展にも一石を投じ得る重要な結果であると考える.

5 Grad-CAM による可視化

近年, ネットワークへの入力に対して, どの部分に注目して予測を行っているかを可視化する技術が注目されている. 本章では, そのような手法のひとつである Grad-CAM [18] を用いた可視化を試みる.

CLCNN によって作られたモデルに対し, Grad-CAM を適用した先行研究に文献 [8] がある. 本稿でもそこでの方法に倣い文字に色付けを行った. 本稿では, ネットワークへの入力文字列に対し, 赤が強くでている文字ほど, 出力を決定時に大きく貢献した文字を意味するようにした. また, 白文字の中でも白が強く出ている文字が赤文字の次に, 出力を決定時に影響を与えた文字であることを意味する.

例えば, 「旅行に行ったら、土産にインフルもらってきた。」を入力したときの結果が Fig. 3a) である. まず, Fig. 3a) の 1,2 行目には, 各「病気/症状」の出力値を示した. また, 出力値の中で 0.5 以上のものがある場合には, 3, 4 行目にその出力値とそのときのクラス番号 (class_idx) を表示するとともに, 5 行目に Grad-CAM で色付けした結果を示した. なお, 0.5 以上の出力値が複数存在した場合には, この 3〜5 行目を複数回繰り返し表示している.

Fig. 3a) から, 「インフルエンザ (Influenza)」と「発熱 (Fever)」の出力値が大きく, インフルエンザが熱と関連付けて学習されていることがわかる. 正解率が 8 割以上なので, ほとんどの例で Fig. 3a) のような学習が行われているが, 例えば, Fig. 3b) のように失敗する例もある.

Fig. 3b) は, 「きつい、上司頭痛の種。」を入力したときの結果である. Fig. 3b) からこの文書では「頭痛の種」に強く反応し, 「頭痛 (Headache)」と判定されていることがわかる. しかし, 「頭痛の種」は, あくまで「頭痛になりそうだ」ということを意味しているに過ぎず, 通常は, 本当に「頭痛」があるわけではないと思われるので, 「頭痛」と判定されるのは好ましくない.

これらの結果から, 「頭痛の種」が「病気/症状」でないことの学習が不十分であったものと予想される. そこで, この違いを分析するために, 学習データ中に, 「頭痛の種」が含まれるツイートを調べたところ, Table. 5 に示した 6 種類存在した. 同じこの Table. 5 には, これらのツイートを入力したときの「頭痛」に対する出力値の一例を同時に示した. Table. 5 の出力値からは, いずれも「頭痛」学習データに対しては適切に「頭痛」でないと学習されていることがわかる.

学習データでは適切に学習されているものの, テストデータでは適切な結果が得られない場合があることの理由のひとつに, 学習データ中に「頭痛の種」と同時に用いられている単語の存在があると思われる. 例えば, Table. 5 の 30ja,519ja,1819ja には「契約」, 503ja,594ja,1228ja には「翻訳」がともに含まれる. 分類に失敗した Fig. 3b) にはこのような Table. 5 の複数のツイートに共通して含まれている単語が含まれていない. それに対し, 「翻訳の出来にうるさい上司。頭痛の種。」や「契約にきつい上司、頭痛の種」のような「頭痛の種」とともに, Table. 5 の複数のツイートに共通して含まれている「契約」や「翻訳」を含む文章

Table 5: 「頭痛の種」が含まれる学習データ

番号	ツイート	頭痛 (Headache)
30ja	この契約が上手くいかないのは頭痛の種でしかない。	0.019387
503ja	この翻訳の量はきつすぎて頭痛の種だ。	0.022169
519ja	今月中に契約を取らなきゃいけない。ほんと頭痛の種だ。	0.01828
594ja	急ぎの翻訳の仕事が舞い込み、土日もつぶれてしまうのが頭痛の種だ。	0.01603
1228ja	いい翻訳ができないことが頭痛の種だ。	0.010696
1819ja	頭痛の種は契約が更新してもらえるかどうかだな	0.056309

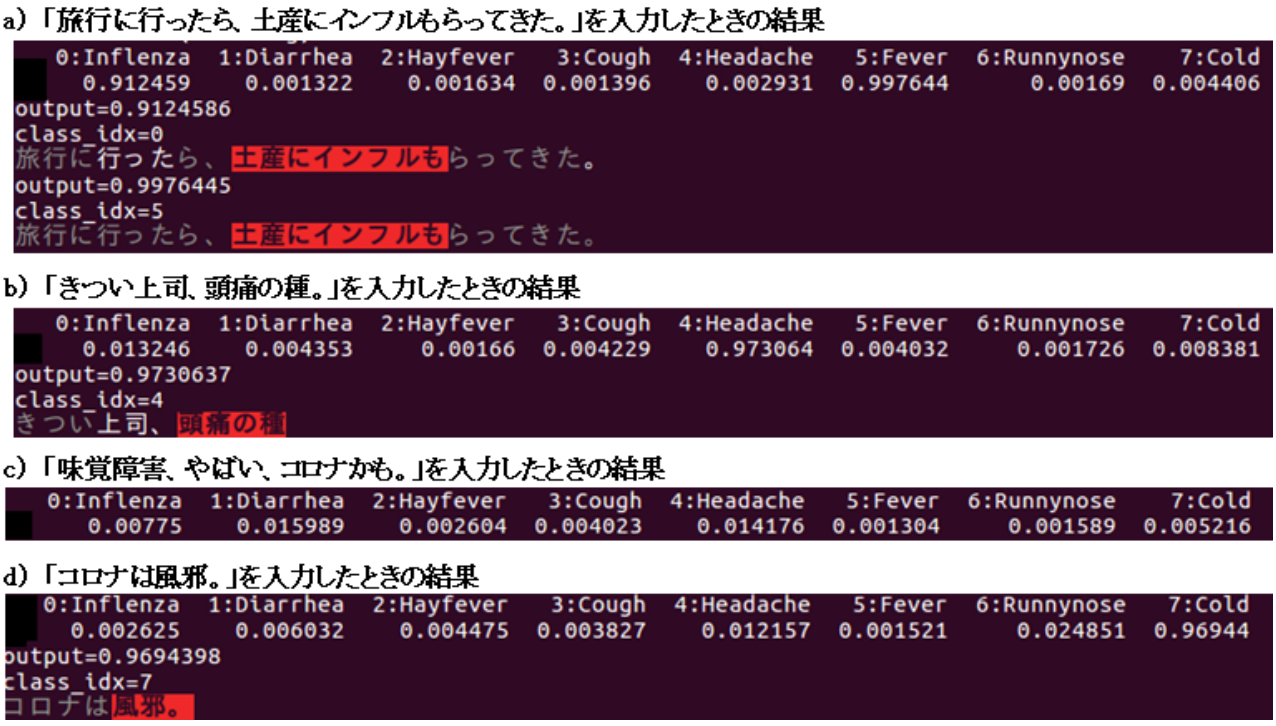


Fig. 3: Grad-CAM の出力結果

を入力したところ, 適切に分類できた. 一方, 「更新のことも頭痛の種である」や「土日もつぶれてしまうのが頭痛の種だ。」のような, Table. 5 の複数のツイートに共通して含まれている単語が含まれていない例では, 「頭痛」と判定されていた. このことからも, 学習データの構成が結果に大きく影響することが Grad-CAN による出力結果を通じて, 確認することができた.

次に, 試しに, 今注目の COVID-19 に関するテストデータを擬似的に作成し入力してみた. 「味覚障害、やばい、コロナかも。」を入力したときの結果が Fig. 3c), 「コロナは風邪。」を入力したときの結果が Fig. 3d) である. 「味覚障害」は NTCIR-13 MedWeb タスクのデータには該当する「病気/症状」が存在しないため, Fig. 3c) に示すようにいずれの「病気/症状」にも該当していない. 一方, Fig. 3d) に示した「コロナは風邪」は「コロナ」が認識できなかったとしても, 「風邪」がキーワードとなるので, 自然に「風邪」に分類されている.

このように未知の「病気/症状」であった場合には, 他に一致するものがあればその「病気/症状」が強くマークされるが, そうでない場合には, いずれの「病気/症状」とも一致しないことが確認できた. これは Grad-CAM により可視化することで, 結果の根拠が人間にとってよく理解できることを意味しており, Grad-CAM が実用性の高い技術であることが確認できた.

6 おわりに

テキスト分析手法として Character-level CNN (CLCNN) が注目されている. 著者らはこれまでに, 大学における 3 つのポリシーの分析に CLCNN を利用し, その有効性を確認してきた. しかしそこでは, マルチクラス問題への適用に留まっており, ひとつのサンプルが複数のクラスに属する可能性のあるマルチラベル問題での有効性は未検証のままであった.

そこで本論文では, マルチラベル問題のテストコレクションとして知られる NTCIR-13 MedWeb タスクを題材に CLCNN の性能評価を行った. 既存手法に対して, 極小解への収束を避ける方法として知られる Dropout や, n-gram を模した畳み込みを行っている本稿の手法の有効性が確認できた. また, ユニット数を増やした実験を通じて, 文献 [1] で紹介されていた定理に従った結果が得られることを確認した. さらに, Grad-CAM を用いることで, とかくブラックボックスと思われがちな深層学習結果の可視化も行った.

今後は, 提案手法を医療診断以外の分野へも適用し, 有効性の検証を行う予定である.

参考文献

1) 甘利俊一. 2014. 新版 情報幾何学の新展開, 第 15 章 深層学習の発展と統計神経力学, サイエンス社.

2) Belkin, M., Dsy, D. & Xu, J. 2019. Two models of double descent for weak feartures, arXiv:1903.07571v1.

3) Bengio, Y., Ducharme, R., Vincent, P. & Jauvin, C. 2003. A neural probabilistic language model, J. of machine learning research, 3, pp.1137-1155.

4) Hastie, T., Montanari, A., Rosset, S. & Tibshirani, R. J. 2019. Surprises in high-dimensional least squares interpolation, arXiv:1903.08560v3.

5) Iso, H., Ruiz, C., Murayama, T., Taguchi, K., Takeuchi, R., Yamamoto, H., Wakamiya, S. & Aramaki, E. 2017. NTCIR13 MedWeb Task: Multi-label Classification ofTweets using an Ensemble of Neural Networks, Proc. of the 13th NTCIR Conference on Evaluation of Information Access Technologies, pp.56-61.

6) Kagaguchi, K., Huang, J. & Kaelbling, L. P. 2019. Effect of depth and width on local minima in deep learning, Neural Computation, 31, pp.1462-1498.

7) Kim, Y. 2014. Convolutional Neural Networks for Sentence Classification, Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing, pp.1746-1751.

8) 北田俊輔, 彌冨 仁. 2018. CE-CLCNN: Character Encoder を用いた Character-level Convolutional Neural Networks によるテキスト分類, 言語処理学会 第 24 回年次大会 発表論文集, pp.1179-1182.

9) Le, Q. V., Ranzato, M., Monga, R., Devin, M., Chen, K. Corrado, G. S., Dean, J. & Ng, A. Y. 2012. Building Highlevel Features Using Large Scale Unsupervised Learning, Proc. of the 29th International Conference on Machine Learning, pp.507-514.

10) Mikolov, T., Chen, K., Corrado, G. & Dean, J. 2013. Efficient Estimation of Word Representations in Vector Space, arXiv:1301.3781.

11) Miyazaki, K. 2017. Exploitation-Oriented Learning with Deep Learning - Introducing Profit Sharing to a Deep Q-Network - , J. of advanced computational intelligence and intelligent informatics, 21, 5, pp.849-855.

12) Miyazaki, K., Takahashi, N. & Mori, R. 2019. Research on Consistency between Diploma Policies and Nomenclature of Major Disciplines: Deep Learning Approach, Proc. of 2019 7th International Conference on Information and Education Technology.

13) 宮崎和光, 井田正明. 2019. Character-level CNN を用いたディプロマ・ポリシーとカリキュラム・ポリシーの整合性判定システムの構築, 電気学会論文誌 C, 139, 10, pp.1119-1127.

14) 宮崎和光. 2020. Character-level CNN を用いた医療用ツイートデータの分類, 計測自動制御学会 システム・情報部門学術講演会 2020.

15) 宮崎和光, 高橋望, 森利枝. 2020. 学位に付記する専攻分野の名称の想起が困難なディプロマ・ポリシーの発見, 計測自動制御学会 システム・情報部門学術講演会 2020.

16) Miyazaki, K. 2020. Classification of Medical Data using Character-level CNN, The 3rd International Conference on Information Science and System, pp.43-47.

17) Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. & Riedmiller, M. 2013. Playing Atari with Deep Reinforcement Learning, NIPS Deep Learning Workshop 2013.

18) Selvaraju, R. R., Das, A., Vedantam, R., Cogswell, M., Parikh, D. & Batra, D.: Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization, CoRR arXiv:1610.02391, 2016.

19) Silver, D. Huang, A.,Maddison, C.J.,Guez, A., Sifre, L., Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T. & Hassabis, D. 2016. Mastering the game of Go with deep neural networks and tree search,Nature,529,pp.484-489.

20) Pennington, J., Socher, R. & Manning, C. 2014. GloVe: Global vectors for word representation, Proc. of the 2014 conference on EMNLP, pp.1532-1543

21) 坪井祐太, 海野裕也, 鈴木潤. 2017. 深層学習による自然言語処理, pp.171-174, 講談社.

22) Wakamiya, S., Morita, M., Kano, Y., Ohkuma, T. & Aramaki, E. 2017. Overview of the NTCIR-13 MedWeb Task, Proc. of the 13th NTCIR Conference on Evaluation of Information Access Technologies, pp. 40-49.

23) http://research.nii.ac.jp/ntcir/permission/ntcir-13/perm-en-MedWeb.html [accessed:2019-11-08].

24) Watkins, C. J. H. & Dayan, P. 1992. Technical note: Qlearning, Machine Learning, 8, pp.55-68.

25) Zhang, X., Zhao, J. & LeCun, Y. 2015. Characterlevel Convolutional Networks for Text Classification, arXiv:1509.01626.

26) https://retty.me [accessed:2019-11-08]