ディープラーニングによるヘイトスピーチ動画の判定

渡邉　雄也† 　宝珍　輝尚† 　野宮　浩揮†

† 京都工芸繊維大学院情報工学専攻　〒 606–8585 京都府京都市左京区松ヶ崎御所海道町
E-mail: †{m6622050,hochin,nomiya}@edu.kit.ac.jp

あらまし　近年，ヘイトスピーチが問題になっており，某動画投稿サイトにもその類の動画が投稿されている．このような動画形式のヘイトスピーチを削除したいという要求があるが，このような動画は，1 時間以上のものもあり，人間が一つ一つチェックして削除対象かを判断するのは莫大なコストがかかる．そこで，ディープラーニングを利用し，自動で動画をヘイトスピーチか否かを判定する手法を提案する．提案する手法では，画像群と音声に対して別々にディープラーニングを行い，それぞれがヘイトスピーチである確率を求め，最終的にその動画がディープラーニングであるか否かを判定する．画像群の判定は畳み込みニューラルネットワークを利用し，音声の判定は感情分野において良好な特徴量セットを利用してディープラーニングを行う．その結果，ヘイトスピーチを精度良く判定できることを示す．

キーワード　ディープラーニング，動画判定，ヘイトスピーチ

1. はじめに

近年，ヘイトスピーチが大きな問題になっており，某動画投稿サイトにもその類の動画が投稿されている．このような動画を削除したいという要求があるが，このような動画は，長くて 1 時間以上のものもあり，かつ，ヘイトスピーチという特性上，好んで見続けられるようなものでもない．つまり，人間が一つ一つ動画をチェックして削除対象か否かを判断するには莫大なコストがかかってしまう．そこで本稿では，ディープラーニングを利用して，自動で動画をヘイトスピーチか否かを判定する手法を提案する．

ディープラーニングは近年，大きく発展している技術である．例えば，2012 年に行われた画像の物体認識のコンテスト (ILSVRC) では，Deep Convolutional Neural Network [1] を用いた手法が，従来の画像認識のアプローチより大幅に性能が向上することが報告された．また，音声認識の分野でも大きな発展を示した [2]．しかし，動画に対するアプローチは確立されていないのが現状である．これは動画が画像群と音声から構成されていることが一因である．また，本稿で取り扱うヘイトスピーチの動画はその特性上，様々な人の声が入り混じった声や文字に表しにくい怒号など音声認識が働きにくいものである．

そこで，本稿では Schuller らによって提案された The INTERSPEECH 2009 Emotion Challenge(以下，IS09 と表記) [3] と The INTERSPEECH 2010 Paralinguistic Challenge(以下，IS10 と表記) [4] で使用された特徴量セットを利用する．これらは人間の感情情報を含めているパラ言語に対する処理を目的としたものであるため，ヘイトスピーチを行う際の感情を認識できるのではないかと考え，使用することにした．また，画像群に関しては，畳み込みニューラルネットワーク（以下，CNN と表記）を利用し，幾つかの画像を入力として与え，学習を施した．最後に，画像群での学習結果と音声での学習結果をクロスバリデーションにより，評価を行い，その結果を示す．

本論文では，まず第 2 章で関連研究について述べ，第 3 章では提案手法を示す．第 4 章で実験設定について述べ，第 5 章で実験結果を示す．第 6 章で考察を行い，第 7 章で結論と今後の課題を述べる．

2. 関連技術

2.1 ディープニューラルネットワーク

2.1.1 活性化関数

ニューラルネットワークでは，各層の出力は活性化関数によって求められる．つまり，各層への入力に重みを掛けたものとバイアスを足したものを活性化関数に与え，出力を行う．この活性化関数は様々なものが存在するが，本稿で使用するシグモイド関数，ReLU 関数とソフトマックス関数について述べる [1], [5]．

まず，シグモイド関数は式 (1) のように表される．

$$sig(x) = \frac{1}{1 + \exp(-x)} \tag{1}$$

この関数がよく使用される一因として，誤差逆伝播法による学習を可能にすることがある．この誤差逆伝播法は出力層よりも前の隠れ層の重みを修正できるものである．この方法では，活性化関数を微分可能な関数に変更することで，出力層での誤差評価への任意の結合重みの寄与を偏微分係数として計算できるようになる．これによって，後述する勾配降下法などの通常の最適化手法によって，出力と正解との誤差が小さくなるように全ての重みを修正することが可能になった．

次に，ReLU 関数について述べる．ReLU 関数は式 (2) のように表される．

$$ReLU(x) = log(1 + exp(x)) \simeq max(0, x) \tag{2}$$

この関数は出力計算，勾配計算が高速に出来る利点がある．また，$max(0, x)$ という性質上，多くの値が 0 になり，出力が疎になり，勾配も 0 になりやすい．そのため，いくつかのニューロンのみが非零であるスパースなニューラルネットワークになる．また，シグモイド関数などの活性化関数はその勾配が多く

の部分で 0 に近く，ディープニューラルネットワーク（DNN）では低い層での誤差が消失する勾配消失問題がある．しかし，ReLU 関数では，DNN であっても誤差は消失せずに伝播する．

最後に，ソフトマックス関数について述べる．ソフトマックス関数は多クラス分類で使用され，K クラスの分類を行う．出力は K 個で，総和は 1 であり，それぞれの出力はそのクラスに属する確率を示す．具体的には，入力 $u_j(j = 1, ..., n)$ をもとに，式 (3) のようにする．ここで，n はクラス数を表す．

$$p_j = \frac{e^{u_j}}{\sum_{k=1}^{n} e^{u_k}} \tag{3}$$

認識時には，p_j が最大値をとるインデックスを推定クラスとする．

2.1.2 畳み込みニューラルネットワーク

画像認識分野において，大きく成功を収めているディープラーニングの手法として CNN がある [1], [5]．CNN の特徴は，畳み込み層およびプーリング層と呼ぶ特殊な層を交互に接続した構造を持つことにある．活性化関数には，ReLU 関数がよく使用される．通常，CNN の出力層間のノード全てを結合する層である全結合層を 1 層以上配置する．そして，CNN の出力を与える出力層は，クラス分類を目的とする場合，活性化関数にソフトマックス関数を用いる．次に，それぞれの層について述べる．

まず，畳み込み層について述べる．畳み込み層への入力は縦横のサイズが $S * S$ 画素の N 枚 (以下，この枚数をチャネルと表記) の画像の形をとる．なお，ここでは簡単のために $S * S$ 画素としているが縦横の画素が一致している必要はない．例を挙げると，縦横が $28 * 28$ 画素のグレースケールの画像を入力に与えれば，$S = 28$, $N = 1$ となり，カラー画像ならば RGB の計 3 枚で $N = 3$ となる．畳み込み層では，この入力にフィルタを畳み込む計算を行う．具体的には，入力のサイズが $S * S$ 画素の各チャネルごとに $L * L$ のサイズの 2 次元フィルタを畳み込み，その結果を全 N チャネルにわたって加算するのが一般的である．このようにして得られた値を活性化関数に入力し，次の層へと入力される．このフィルタの係数は CNN の重みであり，学習によって決定される．

次にプーリング層について述べる．この層は基本的には畳み込み層の出力がこの層への入力となる．従って，プーリング層への入力は $S * S * N$ の形をとる．プーリング層の目的は，画像のどの位置でフィルタの応答が強かったかという情報を一部捨て，画像内に現れる特徴の微小な位置変化に対する応答の不変性を実現することにある．通常，プーリングの処理は画像の縦横方向に間引いて行うため，2 以上のストライド s を設定する．$s = 2$ ならば，出力は入力の縦横半分のサイズとなる．例を挙げると，$s = 2$ で入力画像が $28 * 28$ のサイズならば $14 * 14$ のサイズの画像を出力する．この時に $s * s$ の領域内の画素の平均を出力する方法が平均プーリング，最大値を出力する方法がマックスプーリングと呼ばれる．そのため，プーリング層では，学習する重みは存在しない．

最後に全結合層であるが，これは，隣接層間のノードを全て結合する層であり，特に特徴がある層ではない．

2.1.3 勾配降下法

学習時に現在のパラメタ θ で特徴づけられた目的関数 $L(\theta)$ が与えられた時に，これを最小にする $\theta^* = argmin_\theta L(\theta)$ を求める最適化を行うことで学習を行う．この最適化には勾配降下法を使用するのが一般的である．この勾配降下法には大きく分けて 3 つの方法が存在する．

一つがバッチ勾配降下法と呼ばれるもので単なる勾配降下法である．これは訓練事例全体に対する θ について，コスト関数の勾配を求める．この方法では 1 回の更新を行うのにも全てのデータセットに対して勾配を計算するため，処理速度が著しく遅く，メモリに収まらないデータセットに対して処理はできない欠点がある．

もう一つが確率的勾配降下法 (以下，SGD と表記) である．これはバッチ勾配降下法とは対照的にそれぞれの訓練例 x_i とラベル y_i に対して θ の更新を行う．SGD は訓練事例が冗長である場合に有効である．例をあげれば，全ての訓練事例が複製してサイズが倍となった新しい訓練事例を考えた場合でも，両者から得られた勾配は等しい．このように訓練事例が冗長な場合は，SGD は少ない訓練事例を用いて正確な勾配情報を求めることが出来る．しかし，訓練事例が少数であると，勾配の推定が不安定となり結果として収束が遅くなる．

最後が本稿でも利用するミニバッチ勾配降下法である．バッチ勾配降下法と SGD の中間の策であり，訓練事例を n 個利用しミニバッチ毎に勾配を求めて，この勾配を利用して θ を更新する [5].

2.1.4 Dropout

ニューラルネットは表現力が高いモデルなので過学習が起こりやすい．そこで 2012 年に Hinton らによって過学習を防ぐために考案された手法が Dropout である [6]．Dropout は一定割合 $0 \leqq \alpha < 1$ のニューロンの出力を 0 にして学習を行うものである．例えば，図 1 のように中間層の一部のニューロンを働かせないようにする．また，このように Dropout されたニューロンは，順方向パスに寄与せず，逆伝播にも関与しない．従って，入力が提示されるたびに，ニューラルネットワークは異なるアーキテクチャを構築する．しかし，これらのアーキテクチャは全て重みを共有する．また，ニューロン間の依存関係が減るため，正則化の効果があり，汎化性能の向上を期待できる．通常，Dropout は $\alpha = 0.5$ として利用される．

2.2 openSMILE について

openSMILE ツールキットは信号処理および機械学習アプリケーションのための特徴抽出器である [7]．これは，オーディオ信号機能に焦点が置かれている．本研究では，Schuller らによって提案された IS09 [3] と IS10 [4] で使用された特徴量セットを使用する．これらは人間の感情情報を含めているパラ言語に対する処理を目的としたものである．

IS09 では，声である確率やメル周波数ケプストラム係数（MFCC）などを利用し，それらの最大値，最小値や標準偏差などを特徴量にしており，合計で 384 個もの特徴量がある．IS10 では，IS09 よりもさらに詳細に特徴量を算出しており，合計で 1582 個もの特徴量を利用している．

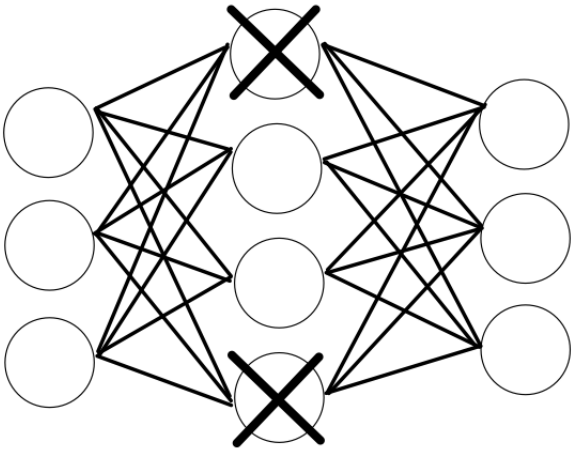

図 1　Dropout の概略図

2.3 画像補間

画像のサイズは全て等しいわけではなく，異なっている場合が多い．その結果，サイズが異なっている画像同士だとモデルへの入力に不具合が生じる場合がある．そのため，様々な画像をある一定のサイズに拡大縮小をしなければならない．しかし，このように拡大縮小をする際には補間方法によっては画像が荒くなりジャギーが発生しやすくなる．

有名な補間方法として，最近隣接補間，バイリニア補間やバイキュービック補間がある [8]．最近隣接補間は拡大か縮小後の画素を拡大率で除算して得た実数値から，その実数値に最も近い画素とするものである．バイリニア補間は同様に得た実数値から，その実数値の周辺 4 画素との距離を算出し重み付けを行うものである．図 2 では，バイリニア補間を表しており黒丸 (拡大か縮小後の画素を拡大率で除算して得た実数値) の画素値は周辺 4 画素により決定される．バイキュービック補間は周辺 16 画素との距離を算出し重み付けを行うものである．これらは，処理速度は最近隣接補間，バイリニア補間，バイキュービック補間の順に高速であるが，同様の順番で画像が荒くなりジャギーが発生したり，ぼやけるようになる．

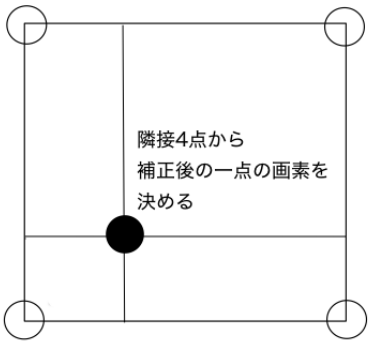

図 2　バイリニア補間の図

3. 提案手法

動画のどの部分がヘイトスピーチであるかを調べるために，本稿では動画を複数に分割する．そして，分割した動画（以下，分割動画と表記）がヘイトスピーチであるか否かを判定する必要がある．また，動画は，画像データ群と音声データによって構成されている．本稿では，これらを別々のデータとして捉えて処理をする．このようにして，分割動画がヘイトスピーチか否かを調べることによって，もととなる動画のどの部分がヘイトスピーチであるかを判定する．図 3 に提案手法の概要を示す．

3.1 動画データの前処理

動画を複数に分割するために FFmpeg [9] を用いる．FFmpeg を用いることによって，任意の動画から一定間隔の複数の分割動画に生成することが出来る (図 4 参照)．

また，動画の音声を処理するために音声データに変換する必要がある．本研究では，afconvert コマンドを用いることによって，分割動画を WAVE ファイルに変換する．この手順によって分割音声データが生成される．

3.2 画像データ群

分割動画は画像データ群から構成される．この画像データ群を全て必要なデータとせずに，それぞれの分割動画の時間軸を中心に N フレーム間隔で M 枚を JPEG 画像として使用する（$N > 0, M > 0$）．この M 枚の画像を DNN に入力として与える．そして，ヘイトスピーチの動画とそれ以外の動画を教師あり学習することによって，ヘイトスピーチを画像群から判定するモデルを生成する．また，学習する分割動画の内訳がヘイトスピーチの動画とそれ以外の動画で異なる場合がある．このような不均衡なデータでは，予測精度が非常に低下する場合があることが知られている．そのため，それらのデータが均衡になるようにサンプリングする必要がある．

3.3 音声データ

ヘイトスピーチの音声は雑音や多数の人の発言のため自然言語処理が難しい．そのため，分割音声データの音響的特徴量を求める必要がある．この音響的特徴量を求めるために openSMILE [7] を使用する．しかし，1 つの分割音声データを全て与えるのではなく，それぞれの分割音声データの最も音が大きい箇所を中心として T 秒間抜き出す（$T > 0$）．これは，音が大きい箇所は最も人間に影響を及ぼすと考えたからである．なお，最も音が大きい箇所が時間軸の 0 秒から $\frac{T}{2}$ 秒に存在する場合は 0 秒から T 秒を抽出する．これは，終わり付近に音が大きい箇所がある場合でも同様と考えたからである．このようにして抽出した音声データから openSMILE を用いて音響的特徴量を求める．この音響的特徴量を DNN に入力として与える．そして，画像データ群と同様に学習を行うことによって，ヘイトスピーチを音声データから判定するモデルを生成する．こちらも画像データ群と同様にデータが均衡になるようにサンプリングする必要がある．

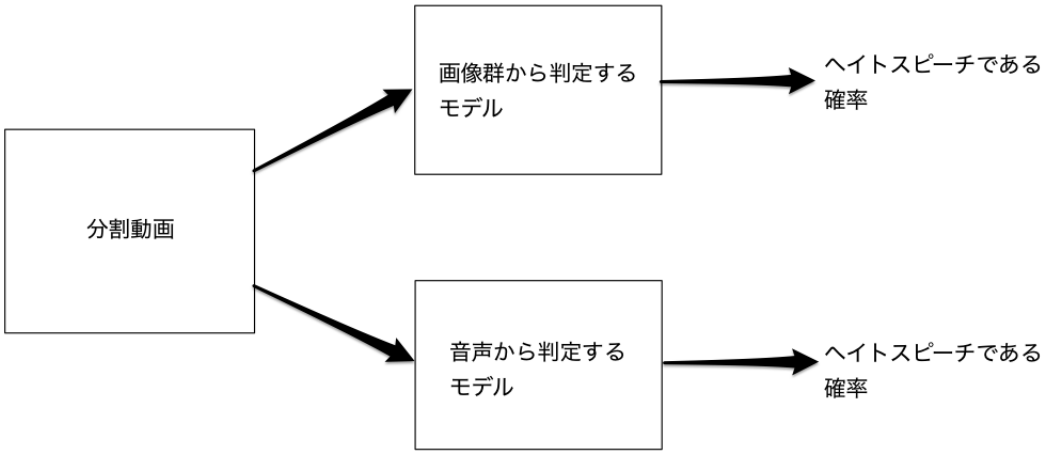

図 3　分割動画判定手法

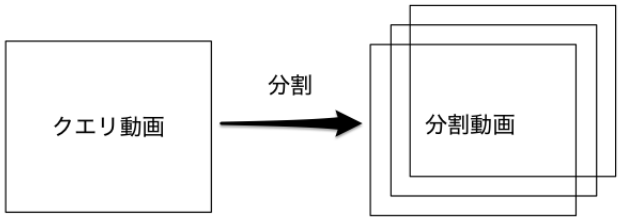

図 4　動画の分割

4. 実験設定

4.1 実験データ

実験データとして，デモ行進や街宣などのデータを 16 個利用した．これらの動画を 1 分ごとに区切ると，分割動画は 614 個となった．これらの分割動画を画像データ群ごとと音声データごとにヘイトスピーチか否かのラベル付けをする．

画像データ群の実験では，ビデオ素材辞典 [10] から 407 個の動画を抜粋し，利用した．これらの動画には，森，川や宇宙などと様々なシチュエーションの動画が存在している．また，音声データの実験では，サウンドバイブル 1200 [11] から 66 個の音声を抜粋し，利用した．これらの音声には，人の笑い声，泣き声や電話音などと様々なシチュエーションの音声が存在している．

4.2 実験方法

画像データ群と音声データでそれぞれの実験方法を述べる．

4.2.1 画像データ群での実験方法

本実験では，分割動画の時間軸を中心に 20 フレーム間隔で 5 枚の JPEG 画像を使用した．つまり，全体で 600 フレームの動画ならば，260, 280, 300, 320, 340 フレーム目を JPEG 画像として生成する．この 5 枚分を 1 つのデータとして使用した．これらの分割動画像を学習用とテスト用に分けた．この時，同じ動画から生成された分割動画は学習データ，もしくはテストデータにまとめて分けた．ここで，学習用データを不均衡データとして扱うことを避ける必要がある．そのため，学習用データの各ラベルの個数をカウントし，個数が多いラベルのデータを個数が少ないラベルのデータの数に合わせた．本実験では，ランダムアンダーサンプリングを用いた．最後に，学習用データを用いてディープラーニングを行い，モデルを作成をし，テスト用データをその作成したモデルに入力として与えた．

DNN の構成は入力層から順に畳み込み層，マックスプーリング層，畳み込み層，マックスプーリング層，全結合層，ソフトマックス層としている．それぞれの層についての詳細を以下に示す．また，本実験で利用している DNN の概略図を図 5 に示す．

- 1 層目の畳み込み層
  - $5 * 5$ フィルタで 32 チャネルを出力
- 2 層目のマックスプーリング層
  - $2 * 2$ のマックスプーリング
- 3 層目の畳み込み層
  - $5 * 5$ フィルタで 64 チャネルを出力
- 4 層目のマックスプーリング層
  - $2 * 2$ のマックスプーリング
- 5 層目の全結合層
  - 1024 の出力
- 6 層目のソフトマックス層
  - 2 クラスの出力

ドロップアウトは全結合層に適用した．また，畳み込み層はパディング部分を 0 で埋めている．重みとバイアスは平均 0，標準偏差 0.1 の正規分布で初期化している．また，入力層ではもとの画像をバイリニア補間によって $28 * 28$ にリサイズしている．なお，ステップ数は 100，バッチサイズは学習データ数を 100 で割ったものとしている．

4.2.2 音声データでの実験方法

本実験では，分割音声データ毎に音が最も大きい箇所を中心に，それぞれ 3，4，5 秒間抽出したデータを使用して実験を行った．それぞれの抽出データをそれぞれの秒数ごとに openSMILE の IS09 と IS10 で使用された音響的特徴量セットに基づいて抽出し，学習用とテスト用に分けた．つまり，3，4，5 秒間抽出

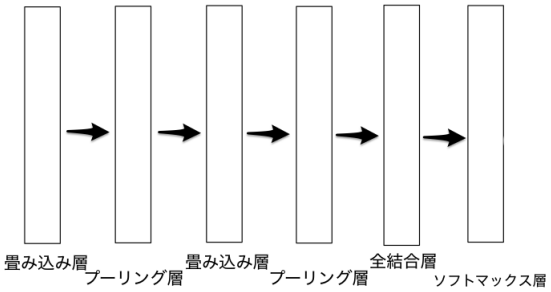

図 5　本実験の画像群の DNN の概略図

したデータの IS09 による特徴量と IS10 による特徴量の合計 6 種類の実験を行った．不均衡データの対処は画像データ群と同様にランダムアンダーサンプリングを用いた．最後に学習用データを用いてディープラーニングを行い，モデルを作成をし，テスト用データをその作成したモデルに入力として与えた．

ディープニューラルネットワークの構成は入力層から順に全結合層，全結合層，ソフトマックス層としている．それぞれの層についての詳細を以下に示す．また，本実験で利用している DNN の概略図を図 6 に示す．

- 1 層目の全結合層
  - 2048 の出力
- 2 層目の全結合層
  - 4096 の出力
- 3 層目のソフトマックス層
  - 2 クラスの出力

ドロップアウトは全結合層に適用し，重みとバイアスは平均 0，標準偏差 0.1 の正規分布で初期化している．なお，ステップ数は 2000，バッチサイズは学習データ数を 20 で割ったものとしている．

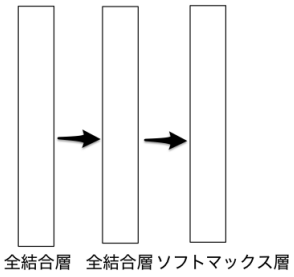

図 6　本実験の音声の DNN の概略図

4.3 評価方法

画像データ群と音声データでの評価はともに leave-one-out 検証法を行った．テスト用データをその作成したモデルに入力として与えた場合の出力の正解ラベルとの合致率を評価対象としている．なお，本稿ではこの合致率を正解率と呼ぶ．また，ヘイトスピーチのデータがヘイトスピーチでないと判断された割合も評価対象としている．また，正解ラベルは著者の中の 2 人によって目視で決定している．これは，ヘイトスピーチのデータがヘイトスピーチでないデータと判断されてしまうことは最も避けられるべきことであるとしたからである．

5. 実験結果

画像データ群と音声データでそれぞれの実験結果を述べる．なお，画像データ群の実験に用いた No.17 以降のものはビデオ素材辞典 [10] のものを表し，音声データの実験に用いた No.17 以降のものはサウンドバイブル 1200 [11] のものを表している．

5.1 画像データ群の実験結果

画像データ群での結果を表 1，表 2 に示す．全体の正解率は 0.94 となった．

表 1　画像データ群の実験結果 (1)

No.	分割動画数	正解数	正解率
1	68	54	0.79
2	22	20	0.91
3	37	37	1.00
4	34	34	1.00
5	44	43	0.98
6	34	30	0.88
7	42	37	0.88
8	21	15	0.71
9	48	45	0.94
10	70	67	0.96
11	38	37	0.97
12	46	44	0.96
13	30	30	1.00
14	29	29	1.00
15	38	38	1.00
16	13	13	1.00
17-423	407	386	0.95
合計	1021	959	0.94

表 2　画像データ群の実験結果 (2)(HS はヘイトスピーチ)

	HS	not HS
HS と予測割合	0.94	0.06
not HS と予測割合	0.06	0.94

5.2 音声データの実験結果

音声データでの結果は複数あるため，IS09 の特徴量セットで最も正解率が高かったものと IS10 の特徴量セットで最も正解率が高かったデータのみを詳細に示す．それが，それぞれ，表 3，表 4 と表 5，表 6 であり，IS09 では 4 秒のデータ，IS10 では 5 秒のデータを利用した方が良い結果となった．また，全体の正解率はそれぞれ 0.74, 0.80 となった．また，IS09 と IS10 で全ての正解率を比較したものを表 7 に示す．

6. 考　　察

画像データ群での結果から，全体の正解率が 9 割以上であり，また，それぞれのクラスの正解率を比較しても，大きな差がないため提案手法で予測を行うことはヘイトスピーチの検索に十分に貢献できると考えられる．しかし，正解率が他の動画

表 3　IS09 で最良だった音声データの実験結果 (1)

No.	分割動画数	正解数	正解率
1	68	50	0.74
2	22	16	0.73
3	37	25	0.68
4	34	31	0.91
5	44	35	0.80
6	34	30	0.88
7	42	24	0.57
8	21	14	0.67
9	48	37	0.77
10	70	59	0.84
11	38	21	0.55
12	46	31	0.67
13	30	27	0.90
14	29	23	0.79
15	38	21	0.55
16	13	7	0.54
17-82	66	55	0.83
合計	680	506	0.74

表 4　IS09 で最良だった音声データの実験結果 (2)(HS はヘイトスピーチ)

	HS	not HS
HS と予測割合	0.75	0.26
not HS と予測割合	0.25	0.74

表 5　IS10 で最良だった音声データの実験結果 (1)

No.	分割動画数	正解数	正解率
1	68	61	0.90
2	22	15	0.68
3	37	30	0.81
4	34	23	0.68
5	44	40	0.91
6	34	28	0.82
7	42	27	0.64
8	21	14	0.67
9	48	46	0.96
10	70	68	0.97
11	38	25	0.66
12	46	30	0.65
13	30	26	0.87
14	29	28	0.97
15	38	23	0.61
16	13	8	0.62
17-82	66	54	0.82
合計	680	546	0.80

と比較して低い動画があるのも事実である．例えば，動画 No.8 などは約 7 割の精度になっている．この動画にある一部の画像データ群は他のデータに類を見ないため，学習が十分にされていないと考えられる．そのため，十分なデータセットを用意することで精度を向上させることが出来るのではないかと考えられる．

表 6　IS10 で最良だった音声データの実験結果 (2)(HS はヘイトスピーチ)

	HS	not HS
HS と予測割合	0.84	0.23
not HS と予測割合	0.16	0.77

表 7　全体の音声データの実験結果

	3 秒間	4 秒間	5 秒間
IS09 での正解率	0.71	0.74	0.68
IS10 での正解率	0.79	0.77	0.80

音声データの結果から，最も高い全体の正解率が 8 割以上であった．IS10 を用いた実験では，3，4，5 秒に大きな差はなかったが，IS10 と IS09 での結果を用いて Welch の t 検定を行うと有意水準 5% で平均値に差があると言える ($p = 2.95 * 10^{-2} < 0.05$)．このことから，少なくとも IS09 より IS10 の特徴量セットを利用した方が推定精度が良好であると考えられる．

以上より，画像群と音声データともにヘイトスピーチか否かを判定することに貢献するモデルが作成できたと言える．

最後に，画像群でのモデルを利用した際のヘイトスピーチである確率と音声データでのモデルを利用した際のヘイトスピーチである確率を動画時間を横軸にグラフ化する．例として，動画 No.1 を入力として与えた場合に画像群でのモデルを利用した動画時間毎のヘイトスピーチの確率の推移を図 7 に示し，音声データでのモデルを利用した動画時間毎のヘイトスピーチの確率の推移を図 8 に示す．これらにより，ユーザは画像群と音声データのそれぞれのヘイトスピーチである確率を視覚的に見ることが出来，ヘイトスピーチである確率が高い動画部分を即時にアクセスすることが出来る．

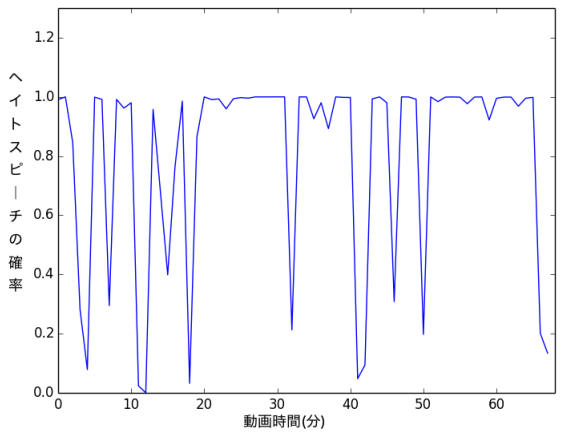

図 7　画像データ群によるヘイトスピーチの確率の推移

7. おわりに

本研究では，任意の動画のどの部分がヘイトスピーチであるか否かを検出する手法を提案した．ここでは，動画に使用されている画像群と音声の双方にディープラーニングを用いた実験を行った．画像群では，CNN を利用することにより良好なモ

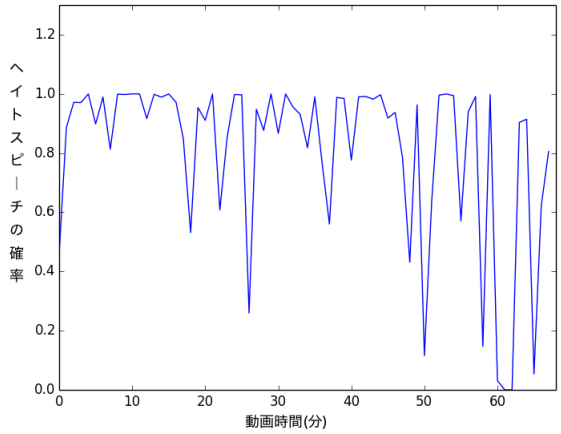

図 8　音声データによるヘイトスピーチの確率の推移

デルを作成できることを示した．また，音声では IS10 の特徴量セットを利用し，それらを学習することにより良好なモデルを作成できることを示した．

今後は，より良好なモデルの模索，より多くのデータの学習実験と GUI システムの構築が課題である．

謝　　辞

本研究を進めるにあたり，データを使用させて頂いた京都府立大学の吉冨康成教授に厚く感謝いたします．

文　　献

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Advances in Neural Information Processing System 25, pp.1097–1105, 2012.
[2] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, “Deep Neural Networks for Acoustic Modeling in Speech Recognition,” IEEE Signal Processing Magazine, pp. 82–97, 2012.
[3] B. Schuller, S. Steidl, A. Batliner, “The INTERSPEECH 2009 Emotion Challenge,” Proceedings of INTERSPEECH 2009, pp.312–315, 2009.
[4] B. Schuller, S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. Muller, S. Narayanan, “The INTERSPEECH 2010 paralinguistic challenge,” Proceedings of INTERSPEECH 2010, pp. 2795–2798, 2010.
[5] 神嶌 敏広, 麻生 英樹, 安田 宗樹, 前田 新一, 岡野原 大輔, 岡谷 貴之, 久保 陽太郎, ボレガラ ダヌシカ, “深層学習,” 近代科学社, pp. 125–188, 2015.
[6] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever and R. R. Salakhutdinov “Improving neural networks by preventing co-adaptation of feature detectors,” Neural and Evolutionary Computing, vol.1207.0580, pp.1–18, 2012.
[7] F. Eyben, M. Wollmer and B. Schuller, “openSMILE - The Munich Versatile and Fast Open-Source Audio Feature Extractor,” Proceedings of the 18th ACM international conference on Multimedia. ACM, pp. 1459–1462, 2010.
[8] G. Bradski, A. Kaebler, “詳解 OpenCV -コンピュータビジョンライブラリを使った画像処理・認識,” オライリージャパン, pp. 133–134, 2009.
[9] FFmpeg, https://ffmpeg.org/（2017/1/10 アクセス）
[10] “ビデオ素材辞典 vol.1-vol.11,” データクラフト, 2002
[11] “サウンドバイブル 1200,” データクラフト, 2009