

# 深層学習を用いた動画のタイトルからタグを予測する 自動ジャンル分け機能

佐口 航 (日本工学院専門学校)

## 目的 (得られる効果)

- ・投稿されたばかりの動画を**視聴者に再生されやすく**する
- ・従来よりも視聴者の**ニーズに合う動画を表示**する
- ・共通するメジャーなタグを**登録する手間を省く**

## 訓練データと正解ラベル (タグ)

訓練データ	タグ一覧
・動画のタイトル	<div> <div>アニメ</div> <div>ゲーム</div> <div>エンタメ</div> <div>音楽</div> <div>スポーツ</div> <div>歌ってみた</div> <div>踊ってみた</div> <div>演奏してみた</div> <div>描いてみた</div> </div> <div> <div>ニコニコ技術部</div> <div>アイドルマスター</div> <div>東方</div> <div>VOCALOID</div> <div>例のアレ</div> <div>生放送</div> <div>政治</div> <div>ニコニコ動画講座</div> <div>静画</div> </div> <div> <div>動物</div> <div>料理</div> <div>日記</div> <div>自然</div> <div>科学</div> <div>歴史</div> <div>ラジオ</div> </div>
正解ラベル	
・動画のタグ	

図1 訓練データと正解ラベル (タグ)

## 開発環境

- ・ Python 3.8
- ・ Anaconda3 (2020.07)
- ・ Spyder 4.1.4
- ・ TensorFlow (GPU) 2.0.0 ※ 1
- ・ Keras 2.3.1
- ・ NVIDIA cuda 10.0
- ・ NVIDIA cuDNN 10.0
- ・ NVIDIA Graphics Driver 441.28
- ・ Microsoft Visual Studio 2017 C++ ※ 1
- ・ Microsoft Windows 10 Pro 64bit 1909

※ 1 KerasをGPUで動作させるため

## データの変換方法

図2のように、String型の文字列を、int型のUnicodeのコードポイントに変換することで機械が学習を行えるようにしました。ベクトル化にvectorize\_sequences関数を、カテゴリ化にto\_one\_hot関数を使いました。

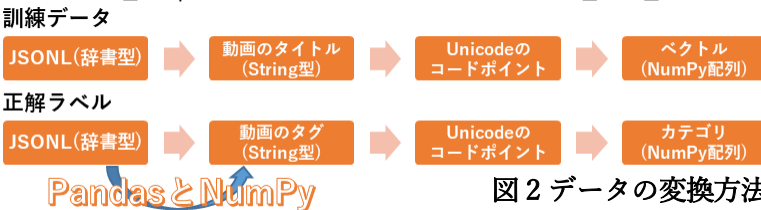


図2 データの変換方法

## 学習モデル(ニューラルネットワーク)の仕様

図3のように、入力層は絵文字に対応するため、Unicodeのコードポイントを10万まで許容したので、ノード数が10万になっています。中間層はPCの性能が許す限りノードとレイヤーを増やしたので、学習モデルのファイルサイズは約825MBになりました。

	ノード数	レイヤー数	活性化関数
入力層	10万個	1層	-
中間層	1,000個	9層	relu
出力層	128個	1層	softmax

表1 学習モデルの仕様1

	コンパイル設定
optimizer	rmsprop
loss	categorical_crossentropy
metrics	accuracy

表2 学習モデルの仕様2

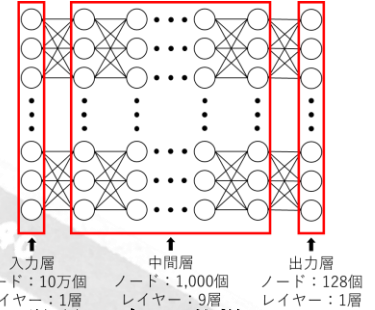


図3 学習モデルの仕様3

## 過学習を防ぐ

過学習とは、規則ではなく答えを覚えてしまう現象です。つまり、過学習が起きるとテストケースでは正解率が高くて、本番環境では正解率が下がってしまいます。これを改善するために、epochsの値に注目しました。epochsとは、「同じ訓練データを何回繰り返して学習させるのか」の回数の事です。図4の損失関数のグラフを見るとepochsの値が15を超えたあたりから、学習の精度の変化が鈍くなることが分かりました。過学習を防ぐために、学習に効果のある回数で止めるため、epochsの値を50から15に下げ過学習を防ぎました。

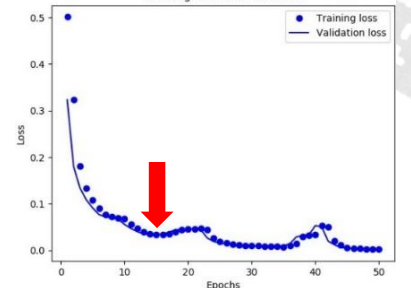


図4 損失関数のグラフ  
ソースコードと学習モデルを公開！  
私的利用の範囲内でご利用頂けます。  
<https://github.com/SaguchiWataru>

## 結果

正解率は、25種類のタグのうち1つ以上登録されている動画のタイトルを出題し、予測されたタグが、実際の動画のタグに含まれているかで正解率を求めた結果、表3のように、約**80.05%**になりました。

出題件数	正解数	正解率
12960354件	10375219件	約80.05%

表3 正解率

## 考察・今後の展望

以上から、動画のタイトルとタグの規則(関係性)を学習させることができました。今回は文字単位で学習を行いましたが、今後は、単語単位の学習に挑戦して精度が上がるのか下がるのか実験してみたいと思いました。特に、"【】"(すみつきかっこ)や"."(ドット)は単語ではないので、どのような結果になるのか興味深く思っております。

## 参考文献

- ・よくわかるPython[決定版]
- ・PythonとKerasによるディープラーニング

## 使用させて頂いたデータセット

ニコニコ動画コメント等データ  
(株)ドワンゴ提供) 2018-12-14 更新



## LT用 既知の問題

投稿された直後の動画は、タグが少ない！



自動でジャンル分け(タグ付け)しよう！

## 重点的に紹介する内容

- ・データの変換方法
- ・ニューラルネットワークの仕様
- ・過学習を防ぐ工夫

## 目的 (得られる効果)

- ・投稿された直後の動画を**視聴者に再生されやすく**する
- ・従来よりも視聴者の**ニーズに合う動画を表示**する
- ・共通するメジャーなタグを**登録する手間を減らす**

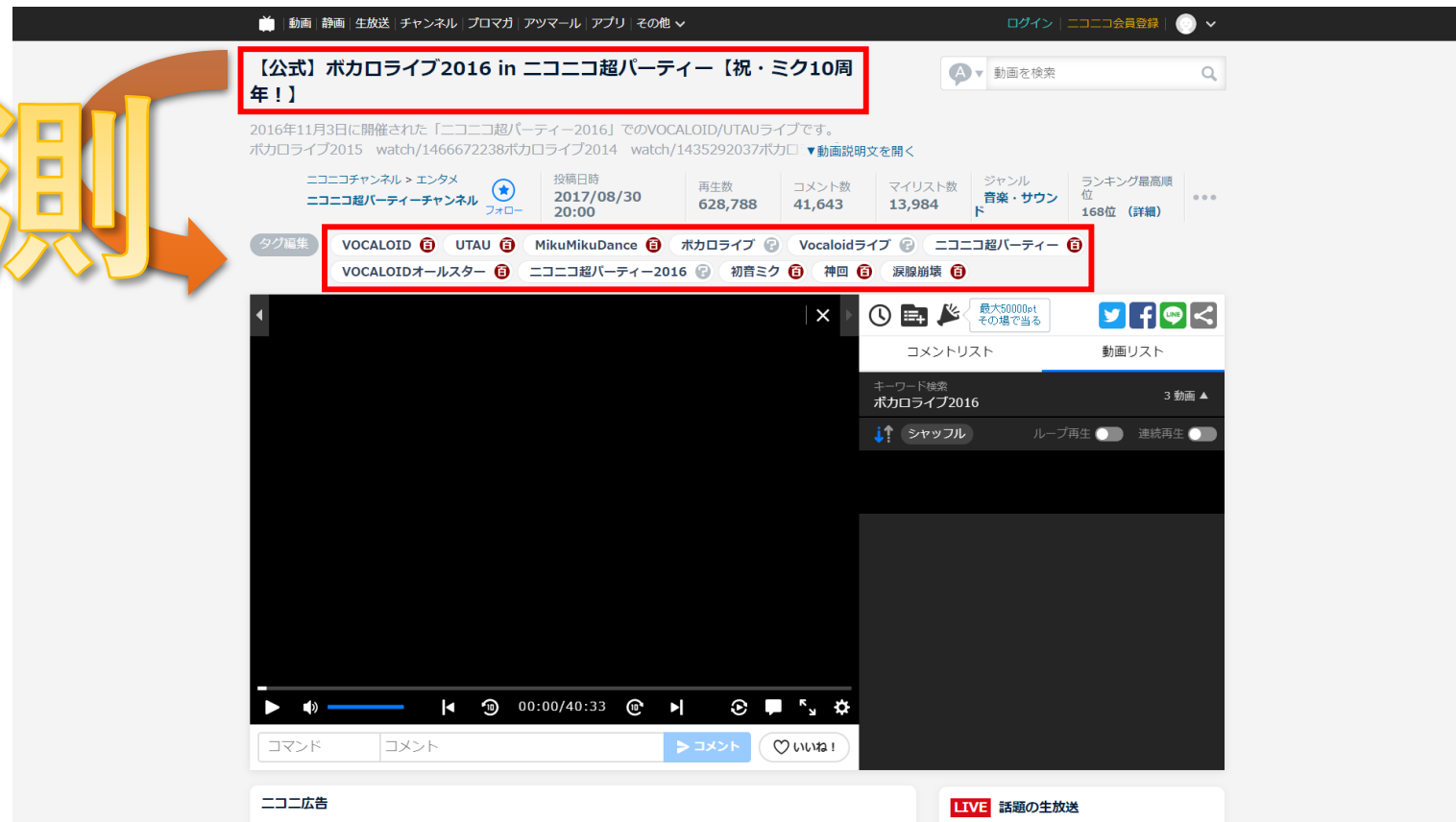
## ソースコードと学習モデルを公開！

私的利用の範囲内でご利用頂けます。

<https://github.com/SaguchiWataru>



予測



Python  
X  
Keras