

2023 年度 修士論文

論文題目

インターネット配信動画における
テキストチャット解析によるハイライト動画生成
(Highlight Clip Video Generation with Text Chat Analysis
for Internet Streaming Video)

指導教員

舟橋 健司 准教授
伊藤 宏隆 助教

名古屋工業大学大学院 工学研究科 工学専攻
情報工学系プログラム 2022 年度入学 34414031 番

上村 岳斗

目次

第 1 章	はじめに	1
第 2 章	背景知識	6
2.1	自然言語処理	6
2.1.1	形態素解析	6
2.1.2	トークン化	6
2.2	ニューラルネットワーク	7
2.2.1	活性化関数	7
2.2.2	損失関数	9
2.2.3	パラメータ更新	9
2.2.4	ファインチューニング	11
2.3	Transformer	11
2.3.1	BERT	13
2.3.2	Sentence-BERT	13
第 3 章	提案手法	15
3.1	提案手法の概要	15
3.2	データセット	15
3.2.1	アノテーション	17
3.2.2	WRIME データセットの利用	19
3.2.3	データセットの前処理	19
3.3	モデルの概要	20
3.3.1	BERT+Linear	20
3.3.2	BERT+pooling	21
3.3.3	BERT+CNN	22
3.3.4	Sentence-BERT	23
3.4	学習の結果	24
第 4 章	評価実験	27
4.1	ハイライト動画生成	27
4.2	被験者による評価実験	27
第 5 章	むすび	35
	謝辞	37

参考文献	38
発表論文リスト	40

第1章 はじめに

近年、情報通信技術の発達や COVID-19 の流行によって人々の生活様式が変化し、自宅で過ごす時間が増えてきている。そのような変化に伴い、映画鑑賞やインターネット上での動画視聴などのデジタルエンターテインメントへの需要が増加している。実際に、2021 年 10 月に行われた調査 [1] では、COVID-19 流行開始前と比べて増えた時間として、「ゲームをする」や、「インターネットで動画を見る」などの自宅で楽しむ娯楽に充てる時間が上位を占めている。特に、「インターネットで動画を見る」に充てる時間については、「13~19 歳」、「20 代」、「30 代」において 50%以上で一位となっている。また、Google の報告 [2] によると、動画配信サービスである YouTube のアクティブユーザー数は、2022 年 1 月の時点で日本国内において 7000 万人を突破したとしている。これらは、COVID-19 の影響や情報通信技術の発達によって、日本国内においてインターネット上で動画を見ることの需要が非常に高いことを示している。

さて、インターネット上における動画配信サービスではライブ配信を行うことができる場合がある。ライブ配信は視聴者に対してリアルタイムな臨場感を提供することができる仕組みであり、その配信コンテンツには視聴者と配信者との対話型コンテンツやゲームプレイ、音楽などのライブパフォーマンスイベントなどがある。これらは、リアルタイムで進行するため、視聴者にとってより感情移入しやすいコンテンツだといえる。また、動画配信サービスでは、様々な理由でライブ配信を視聴できなかったユーザーや、ライブ配信をもう一度視聴したいと考えているユーザーに向けてライブ配信終了後にもライブ配信の動画を視聴できるようにアーカイブ（一般に配信アーカイブと呼ばれている）を提供している。しかし、ライブ配信は数時間から十数時間実施される場合があり、配信アーカイブを利用してもライブ配信の内容をすべて視聴することは難しい。そのため、ライブ配信におけるハイライト部

分のみを抽出したハイライト動画の需要が高まっている。ハイライト動画を作成するには一度配信アーカイブもしくはライブ配信を視聴し、ハイライト部分を見つけなければならない。先に述べたように、ライブ配信は長時間行われる場合があり、そのような場合にはライブ配信の内容をすべて視聴することは時間と労力を費やすことになる。そこで、ライブ配信動画から自動的にハイライト動画を生成するための研究が行われている。

Wei-Ta Chu らの研究 [3] では、リアルタイムで進行するゲームコンテンツのビデオに焦点を当て、ゲーム内で重要なイベントが発生した際に画面に表示されるあらかじめ指定されたテキストを認識することによってゲーム内のイベントを検出している。また、ビデオフレーム間のモーション強度やゲームプレイ画面内に表示される、各ゲームプレイヤーのアバターの数などの視覚的な外観の特徴を利用し、ハイライト動画を自動的に生成するモデルを提案している。ハイライト動画の自動生成を目的とした研究では、このように動画内の視覚的、音声の特徴を利用するケースが多い。

ところで、ライブ配信では視聴者がライブ配信の内容へのリアクションとしてテキストを書き込むことのできるテキストチャット機能が提供されている場合がある。動画配信サービスでは、テキストチャットが図 1.1 の様にライブ配信画面の右側に配置されている場合が多く、ライブ配信の視聴者及び配信者の両方が視認可能な状態で表示されることが多い。テキストチャットはライブ配信の内容に対して視聴者によって書き込まれているため、ライブ配信内の出来事に対してのリアクションをリアルタイムで表現していると考えられる。すなわち、ライブ配信においてハイライト動画に含まれるようなハイライト部分には、それに対するリアクションを表すテキストチャットが多く書き込まれていると予想できる。実際にテキストチャットを特徴として利用し、ハイライト動画の自動生成を試みている研究 [4] がある。この研究では、長時間の動画から手作業で作成されたハイライト動画を学習用データとして、機械学習によるハイライト動画を自動生成することを検討をしている。なおインターネットの世界では、ライブ配信動画をはじめ、長時間の動画を対象に、短時間で要点を視聴できるようにまとめられたハイライト動画は切り抜き動画と呼ば



図 1.1: 日本テレビ [日テレ NEWS], (2023, 12, 30), 【朝ニュースライブ】最新ニュースと生活情報 (12月30日)——THE LATEST NEWS SUMMARY(日テレ NEWS LIVE), https://www.youtube.com/watch?v=WUGdTqmt_S8. (Accessed on 01/06/2024).

れている。実験ではライブ配信中に視聴者の書き込んだテキストチャット数が多い部分が切り抜き動画に含まれていたか調査した。結果として、切り抜かれた区間は書き込まれたテキストチャット数が多かったものの、ライブ配信内におけるテキストチャットの書き込み数が最大の区間ではなかったことを示している。

また、植野らの研究 [5] では、ライブ配信動画中に書き込まれたテキストチャットの内3%をランダムで抽出し、著者自身がそのテキストチャットが含まれる動画シーンを視聴し、そのシーンが「面白い」か「面白くない」かの二択で評価することで、テキストチャットのアノテーションを行った。さらに、アノテーションを行ったテキストチャットを用いて事前学習済みの機械学習モデルをファインチューニングすることで、著者が事前に面白いシーンとして決定していたシーンを含むハイライト動画の生成を行えたとしている。研究では今後の課題点として、著者が見たいシーンのハイライトの生成に成功したが、特定の個人が見たいと感じるシーンのハイライトの生成に限定していることと、ハイライト動画生成の対象となる人が毎回アノテーションを行い、そのデータを用いてモデルのファインチューニングを行わなければならないことを挙げている。

そこで本研究では、YouTube で行われたゲームプレイのライブ配信中に視聴者か

ら書き込まれたテキストチャットを用いて機械学習を行い、作成した機械学習モデルを用いることで、特定の個人に向けたハイライト動画ではなく、多くの人がライブ配信のハイライト動画として良いと感じるハイライト動画の自動生成を目指す。書き込まれたテキストチャットの内容を機械学習モデルにより判断することで、単純にテキストチャットの書き込み数の多い場所を抽出する手法では抽出できなかったハイライト部分の抽出を試みる。加えて、人気のあるハイライト動画を機械学習用データのアノテーションに利用することで、特定の個人ではなく多くの人がハイライト部分だと感じる箇所を抽出して、同ジャンルの動画に対しては、ハイライト動画作成毎にデータのアノテーションを行う必要なしに、ハイライト動画の自動生成を実現する。具体的には、まず、ゲームプレイのライブ配信を行っている人気のYouTubeユーザー1名と、このユーザーのライブ配信動画からハイライト動画を作成しているユーザーの内、YouTubeにおけるチャンネル登録者数が1万人以上のユーザー2人の計3人を選出した。人気のあるハイライト動画作成者のハイライト動画は、ライブ配信動画におけるハイライト部分をハイライト動画に含むことが出来ていると仮定し、ライブ配信中に書き込まれたテキストチャットと作成されたハイライト動画を用いて、テキストチャットに対してハイライト動画に含まれる区間に書き込まれたもの（正解）か、それ以外の区間に書き込まれたもの（不正解）か、アノテーションを行った。アノテーションしたテキストチャットを用いて、事前学習済みの機械学習モデルをテキストチャットがハイライト動画に採用されていた区間に含まれるかどうかを2値分類するタスクによってファインチューニングを行った。次に、ファインチューニングしたモデルを用いて、ハイライト動画生成の対象とするライブ配信中に書き込まれたテキストチャットの2値分類を行う。そして10秒を1区間とし、モデルが正解と分類したテキストチャットが多い区間順に、その区間と直前の2区間と合わせた合計30秒の区間をハイライト動画に含める区間とする。ここで、前の2区間もハイライト動画に含めるのは、テキストチャットの入力の手間やタイムラグを考慮し、ハイライト部分をハイライト動画に含めるためである。また、採用する区間が重複した場合は一つの区間にまとめる（例： $[10,40],[30,60] \rightarrow [10,60]$ ）。最後に、あらかじめ設定した閾値にハイライト動画に含める区間数が達

した際に、ハイライト動画に含める区間の決定を終え、ハイライト動画の作成元にしたライブ配信の時系列に合うように区間を並べ替える。このようにして逐次的にハイライト動画の自動生成を行った。本研究では、システムが自動生成したハイライト動画を、選定したユーザーのゲームプレイのライブ配信動画を視聴したことがあると回答した被験者に視聴してもらい、自動生成したハイライト動画を評価してもらうことでシステムの評価を行った。

本文では、第2章で本研究で用いた自然言語処理や機械学習に関する背景知識について述べる。第3章では、配信アーカイブからハイライト動画を自動生成するための提案手法の詳細について述べる。第4章では、被験者実験を実施し、結果と考察について述べる。第5章では、本研究のまとめと今後の課題について述べる。

第2章 背景知識

本章では背景知識として、本研究に用いた自然言語処理と機械学習について述べる。

2.1 自然言語処理

自然言語とは、人々が日常的に使用している言語であり、自然言語処理とは自然言語情報をコンピュータが認識できる形式に変換し、解析や処理を行うための技術である。自然言語処理によって、チャットボットや文章の感情分析、文章の内容の自動的な要約などが可能になった。特に近年、深層学習技術の発達によって ChatGPT のように、より高度な自然言語の解析が可能となっている。本研究では、ライブ配信中に視聴者が書き込んだテキストチャットの解析に自然言語処理を用いている。

2.1.1 形態素解析

形態素解析とは、自然言語を形態素という単位に分割する操作であり、自然言語処理で最初に行う操作である。例えば「私は人間です」という文章は、「私」、「は」、「人間」、「です」という4つの形態素に分割される。本研究では、オープンソース形態素解析エンジンである MeCab[6] を用いて形態素解析を行っている。

2.1.2 トークン化

トークン化は、文章や文を形態素より小さな単位であるトークンに分割する処理である。トークンは通常、文章内の単語や句読点、数字など、処理対象として意味を持つ最小の単位であり、トークン化は、形態素解析とは異なり、文字列をトークンに分割する処理を指す。トークン化には後述する事前学習済みの機械学習モデルを利用している。また、各トークンは、モデルが学習した語彙に基づいて一意の ID に変換される。具体的には、「私は人間です」という文章はトークン化によって二つの

特殊トークンを加えた「[CLS], 私, は, 人, 間, です, [SEP]」というトークンに分割され, それぞれ「2, 1325, 9, 53, 284, 2992, 3」という ID に変換される. この処理により, コンピュータは自然言語を数値的に判断し, 処理を行うことが可能になる.

2.2 ニューラルネットワーク

ニューラルネットワークとは, 人間の脳の神経細胞を模して考案された数学モデルである. ニューラルネットワークは複数の層から構成され, ニューラルネットワークへの入力が行われる層を入力層, 出力が行われる層を出力層, 入力層と出力層の間に存在する層を隠れ層と呼ぶ. 特に, 隠れ層が複数存在するニューラルネットワークをディープニューラルネットワーク (DNN) と呼ぶ. ニューラルネットワークにおける各層は複数ノードから構成されており, 各層のノード間の結合には重みがある. また, 各ノードにおいて入力の合計に加算される定数 (バイアス) があり, バイアスによってニューラルネットワークが非線形性を学習できる. ニューラルネットワークにおける学習とは, 出力層の出力が目標の出力となるよう重みとバイアスのパラメータを更新することである. 学習を繰り返すことにより, 入力から目標の出力を生成する精度が向上する. 入力を x , 出力を y , 重みを W , バイアスを b とした場合のニューラルネットワークの模式図を図 2.1 に示す.

2.2.1 活性化関数

活性化関数とは, 各ノードへの入力の総和から出力を決定するための非線形な関数である. この関数の非線形性によって, ネットワークは入力されたデータにおける, より複雑なパターンや特徴を学習できる. 活性化関数への入力とその出力を具体的に示すため, 図 2.1 における h_0 のノードに注目した模式図を図 2.2 に示す. h_0 には二つのノード x_0 と x_1 がつながっており, それぞれのノードに対する重みは W_{00} , W_{01} であり, バイアスは b_0 である. 活性化関数を f としたとき, h_0 の出力 $output$ は以下の式 2.1 であらわされる.

$$output = f(w_{00}x_0 + w_{01}x_1 + b_0) \quad (2.1)$$

本研究では, 2種類の活性化関数を用いている. 作成したニューラルネットワークの

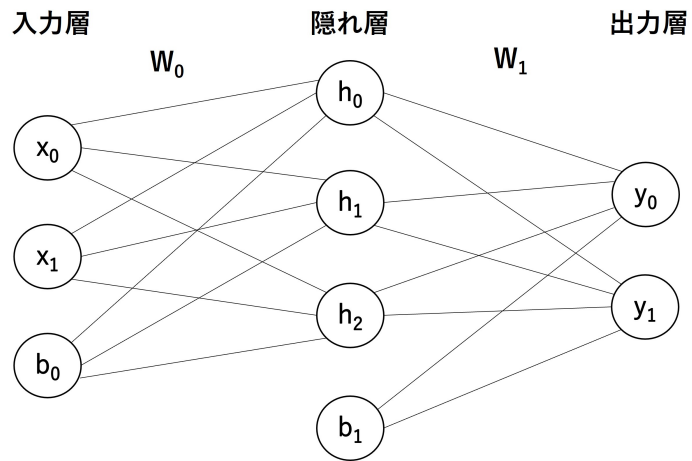


図 2.1: ニューラルネットワークの模式図

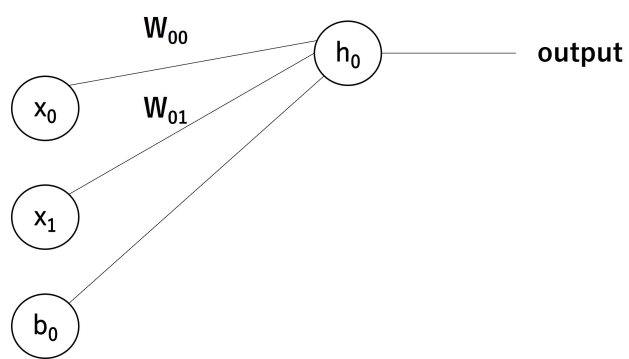


図 2.2: ニューラルネットワークの模式図

出力層に Sigmoid 関数を、畳み込みニューラルネットワーク (CNN) の内部で ReLU 関数を用いている。それぞれの関数について、入力を x 、出力を y として以下の式 2.2, 式 2.3 に示す。

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (2.2)$$

$$\text{ReLU}(x) = \max(x, 0) \quad (2.3)$$

また、各関数の概形を図 2.4, 図 2.3 に示す。それぞれの図より、各関数が非線形であることが確認できる。

2.2.2 損失関数

損失関数とは、機械学習において、ニューラルネットワークの出力と実際の目標値との差異や誤差を評価するための関数である。モデルの学習を行う際には、この損失関数の出力値を最小化するようにパラメータを調整する。損失関数は、ニューラルネットワークの出力値が目標にどれだけ近いかを示し、最適なパラメータ設定を探索するために用いられる。本研究では、損失関数に Binary Cross-Entropy Loss (BCE) を使用する。ニューラルネットワークの出力層の出力を \hat{y} 、出力させたい目標の値を y とすると、BCE は以下の式 2.4 で表される。

$$\text{BCE}(y, \hat{y}) = -(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})) \quad (2.4)$$

2.2.3 パラメータ更新

本研究では、ニューラルネットワークのパラメータ更新にミニバッチ勾配降下法を用いる。ミニバッチ勾配降下法とは、データセットをミニバッチと呼ばれる一定のサンプル数に分割し、各ミニバッチごとに損失を最小化する勾配を計算し、パラメータの更新を行う手法である。また、パラメータを W 、パラメータの更新の量を決める値である学習率を ϵ 、損失を E とすると、パラメータ更新は以下の式 2.5 で表される。

$$W = W - \epsilon \frac{\partial E}{\partial W} \quad (2.5)$$

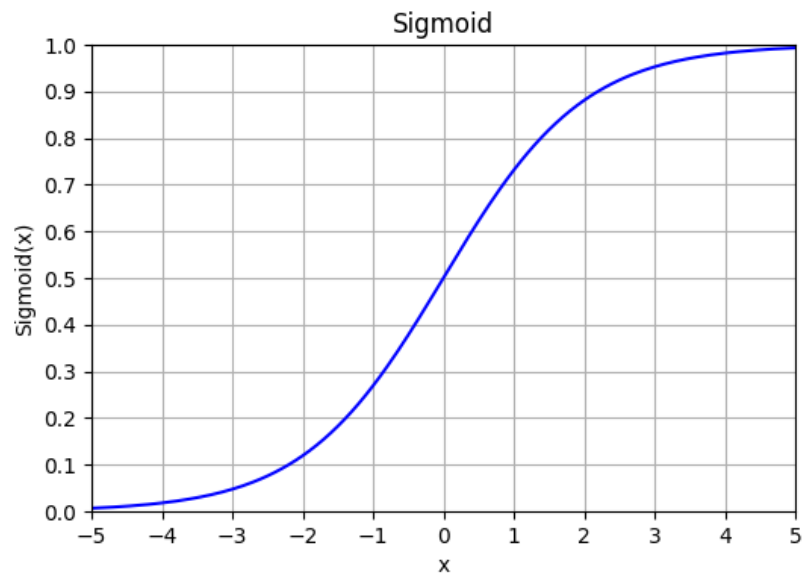


图 2.3: Sigmoid 関数概形

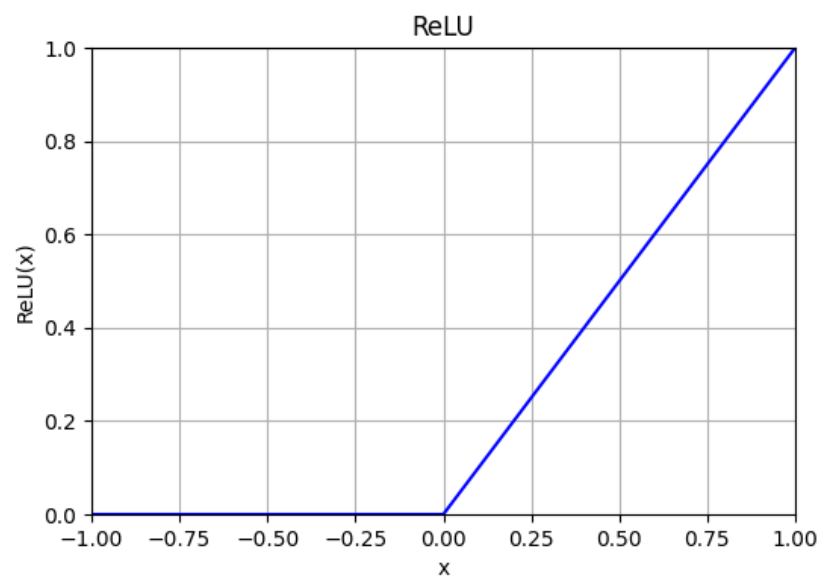


图 2.4: ReLU 関数概形

本研究では、学習率の初期値は 1×10^{-3} とし、最適化アルゴリズムである Adam[7] を利用して、学習率の最適化を行った。

2.2.4 ファインチューニング

ファインチューニングとは、あるタスクを事前に学習済みのモデルを別の新しいデータセットやタスクに適応させる手法である。具体的には、事前学習済みモデルを、新しいタスクに対応するデータセットを用いて追加の学習を行い、モデルの一部や全体のパラメータを調整する。ファインチューニングは少ないデータセットでも効果的なモデルを作成することができる。本研究では、後述する事前学習済みモデル BERT[8] を用いてファインチューニングを行った。

2.3 Transformer

Transformer[9] は、2017年に発表された翻訳を目的とした、自然言語処理のための機械学習モデルである。モデルには Attention 機構 [10] が用いられており、モデルが入力データの特定の部分に注目することが可能である。Transformer の構造を図 2.5 に示す。Transformer は入力された文章（トークン）を特徴量を表すベクトルに変換するエンコーダ（図左半分）とベクトルから文章（トークン）を出力するデコーダ（図右半分）の両方を搭載している、エンコーダ-デコーダモデルである。Transformer は、2つの同じ形態素が文章内の異なる位置に存在する場合に、それぞれ同じベクトルに変換されることを避けるため、Positional Encoding を導入している。文章内における各トークンの位置を考慮することで、同じトークンでも各位置に対して異なるベクトルに変換され、各トークンの位置情報を考慮できる。また、Self-Attention 機構と Multi-Head Attention 機構が導入されており、各トークンが文章内の他のトークンとどれだけ関連しているかを計算することができる。このメカニズムにより、長文における各トークン間の依存関係や文脈を捉えることができ、文章全体の情報を考慮することが可能である。これらの特徴により Transformer は特に自然言語処理のタスクにおいて優れた性能を発揮している。

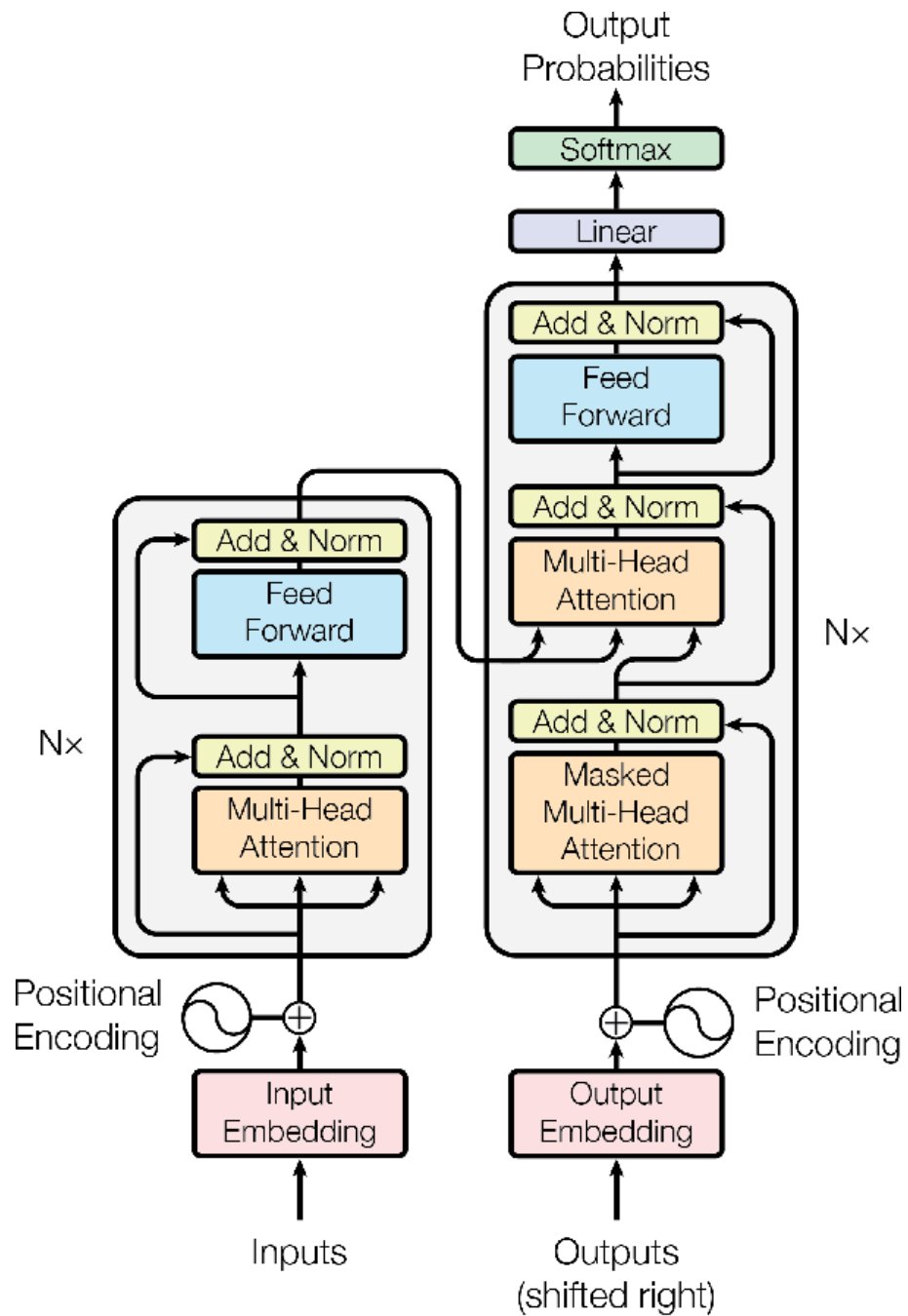


図 2.5: Transformer の構造 (文献 [9] より引用)

2.3.1 BERT

BERT(Bidirectional Encoder Representations from Transformers)[8]は、2018年に発表された、自然言語処理のための事前学習済みモデルである。BERTは先述したTransformerのエンコーダを用いて実装されている。BERTの学習は大規模なコーパスを用いて、マスクされた部分の単語を当てるというタスクによって行われている。BERTの構造を図2.6に示す。BERTへの入力、トークンの埋め込み表現(図中E)である。図中のTrmは2.3節で述べたTransformerのエンコーダである。また、従来の言語モデルでは、文章を左から右あるいは右から左に単語を予測する単方向の学習が行われていたが、BERTでは、Transformerのself-Attention機構を利用して各トークンと他のすべてのトークンとの関連性を計算し、文脈を単方向ではなく、双方向で理解することができる。本研究では、東北大学の乾研究室が開発した、日本語のデータセットを用いて学習を行った事前学習済みモデル(日本語BERT)[11]を用いて機械学習モデルを作成し、モデルのファインチューニングを行うことで、入力されたテキストチャットの2値分類を行う分類器を作成した。

2.3.2 Sentence-BERT

Sentence-BERT[13]は、BERTの出力にPooling演算を追加したモデルで、BERTとは異なり、固定サイズの文章の特徴量を算出する。Sentence-BERTは入力された文章の類似性に関する3種類のタスクを学習することにより、文章の意味的な表現を獲得し、異なる文章の特徴量を比較可能な形で生成する。これにより、文章の意味的な類似性や関連性を効果的に捉えることができる。Sentence-BERTでの文章の類似度の計算の概要を図2.7に示す。本研究では、日本語のデータセットを学習したSentence-BERT[14]をテキストチャットの類似度の計算に利用した。

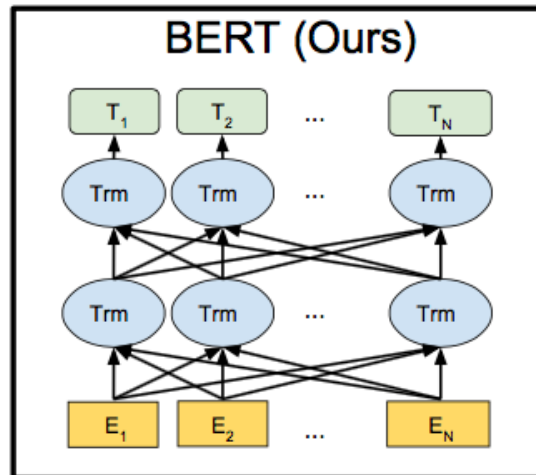


図 2.6: BERT の構造 (文献 [8] より引用)

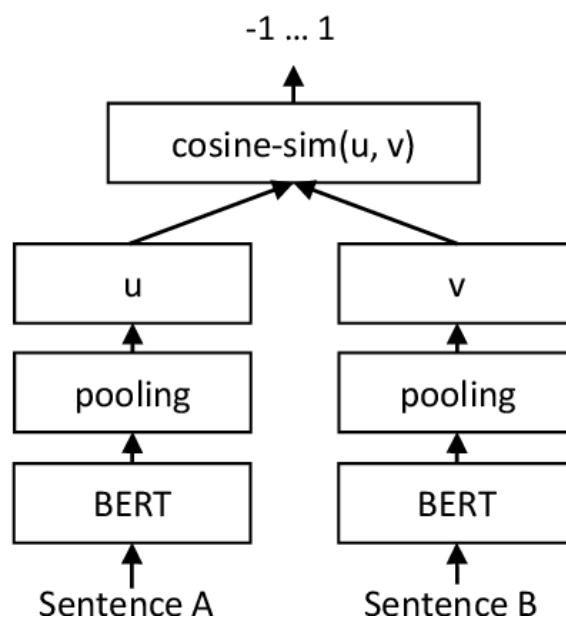


図 2.7: Sentence-BERT による類似度の計算 (文献 [13] より引用)

第3章 提案手法

本章では，ライブ配信中に視聴者が書き込んだテキストチャットを利用して，ハイライト動画を自動生成するための手法の詳細について述べる．

3.1 提案手法の概要

提案手法の流れを図 3.1 に示す．本研究では，ライブ配信中に書き込まれたテキストチャットが，ライブ配信においてハイライト動画に含まれている部分に書き込まれたものかどうかの2値分類を行う機械学習モデルを，日本語 BERT のファインチューニングによって作成する．次に，10秒を1区間として，モデルがハイライト区間に含まれると判定したテキストチャットが多い区間順に，その区間と前2区間と合わせた3区間をハイライト動画に含める区間として決定し，あらかじめ設定した閾値に決定した区間数が達したところでハイライト動画に含める区間の決定を終える．最後に，各区間を元のライブ配信動画と同じ時列に合うよう並び替えることでハイライト動画を自動生成する．

3.2 データセット

ファインチューニングを行うためのデータセットには，ライブ配信動画，同一のライブ配信中に書き込まれたテキストチャットと書き込まれた際の時間のデータ，同一のライブ配信から作成されたハイライト動画を用いる．ライブ配信のジャンルは，イベントが発生しやすく，視聴者から書き込まれるテキストチャット数が多いと考えられることからゲームプレイとし，プレイしているゲームはシューティングゲームの VALORANT とした．次に，利用するハイライト動画について述べる．ハイライト動画を作成しているアカウントの内，一定以上の人気があるアカウントは，多くの視聴者がライブ配信におけるハイライト動画として適当だと感じているハイラ

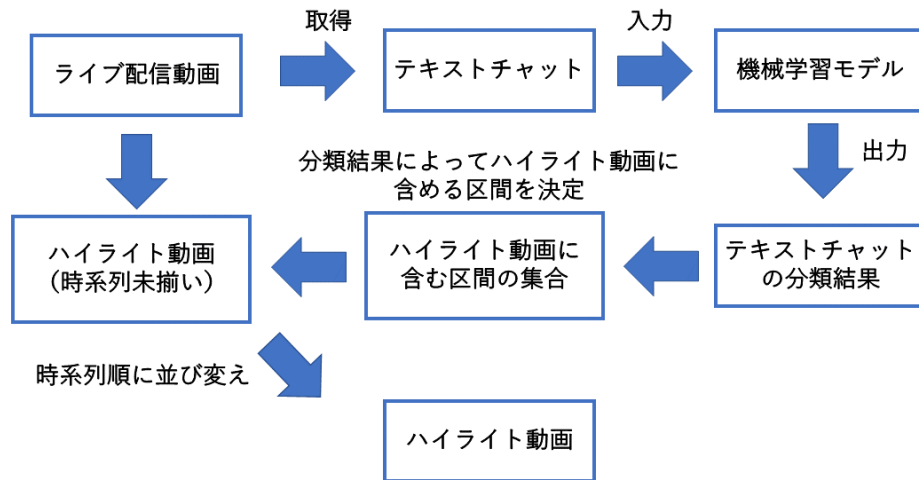


図 3.1: 提案手法の概要図

イト動画を作成できていると仮定し、同一のライブ配信動画を元に作成したハイライト動画を投稿したユーザーで、YouTubeにおける「チャンネル登録者数」が1万人を超えている2つのアカウントが投稿したそれぞれのハイライト動画とした。また、用いるハイライト動画は、ライブ配信内の特定の1つのハイライト部分を用いて作成したものではなく、ライブ配信内の全体から複数のハイライト部分を用いて作成されたものを対象とする。以上を考慮し、提案手法に利用するデータセットとしてANYCOLOR株式会社に所属し、YouTubeでライブ配信を行っている「葛葉」が行ったライブ配信動画及び、そのライブ配信中に書き込まれたテキストチャットと書き込まれた際の時間のデータ、葛葉が行ったライブ配信動画のハイライト動画の作成、投稿を行っている2人のユーザー「葛葉切り抜きch」、「葛葉切り抜き集」が同一のライブ配信動画に対してそれぞれ作成したハイライト動画を用いた。使用したデータを表3.1に示す。ライブ配信動画Aの長さは3時間21分22秒で取得したテキストチャットの総数は18730件である。用いた3つの動画はテキストチャットのアノテーションに利用し、アノテーションを行ったテキストチャットをモデルのファインチューニングに利用した。

表 3.1: 学習用に用いたデータ

ユーザー名	使用したデータ	動画 URL
Kuzuha Cannel	ライブ配信動画 A テキストチャット	https://www.youtube.com/watch?v=Jm0Gwt-XjzI
葛葉切り抜き ch	ハイライト動画 (A を元に作成)	https://www.youtube.com/watch?v=_QHmd2xrTTs
葛葉切り抜き集	ハイライト動画 (A を元に作成)	https://www.youtube.com/watch?v=r-nzZbr9dhA

3.2.1 アノテーション

テキストチャットのアノテーションの方法を図 3.2 に示す. 3.2 節で述べたライブ配信動画 A と A を元に作成された 2 つのハイライト動画を用いて, ライブ配信動画全体 (図中青軸) における, ハイライト動画に採用されていた区間 (図中赤軸) のテキストチャットをラベル 1 (正解), それ以外をラベル 0 (不正解) としてアノテーションを行った. しかし, このようにしてアノテーションを行ったテキストチャットを用いてファインチューニングを行ったところ, 入力されたテキストチャットを, 実際の正解不正解に関わらず, すべて正解と分類するモデルができた. これは, ハイライト動画として採用されなかった区間にも, ハイライト部分が存在するため, 不正解データ内に, 正解データと似た表現のテキストチャットが混在してしまうことが原因だと考えられる. また, ハイライト動画では, ライブ配信動画内のハイライト部分のみでなく, なぜハイライト部分内のイベントが発生したのかを理解するために, ハイライト部分の直前数秒から数十秒を含んでいる場合が多く, 図 3.2 の手法でのアノテーションでは, ハイライト部分に対するものでないテキストチャットを正解としてしまっており, そのことが影響を及ぼした可能性がある. そこでこれらの問題を改善するため, アノテーション手法の改善を行った.

今回対象としたハイライト動画では, ライブ配信動画におけるハイライト部分のみではなく, ハイライト部分が発生した原因が理解できるよう, その前の部分をハイライト動画に含めていた. そこで, ハイライト動画内におけるハイライト部分に対して書き込まれたテキストチャットのみを正解とするようアノテーションを行った. まず, 3.2 節で述べた 2 つのハイライト動画には, ハイライト動画に採用した各

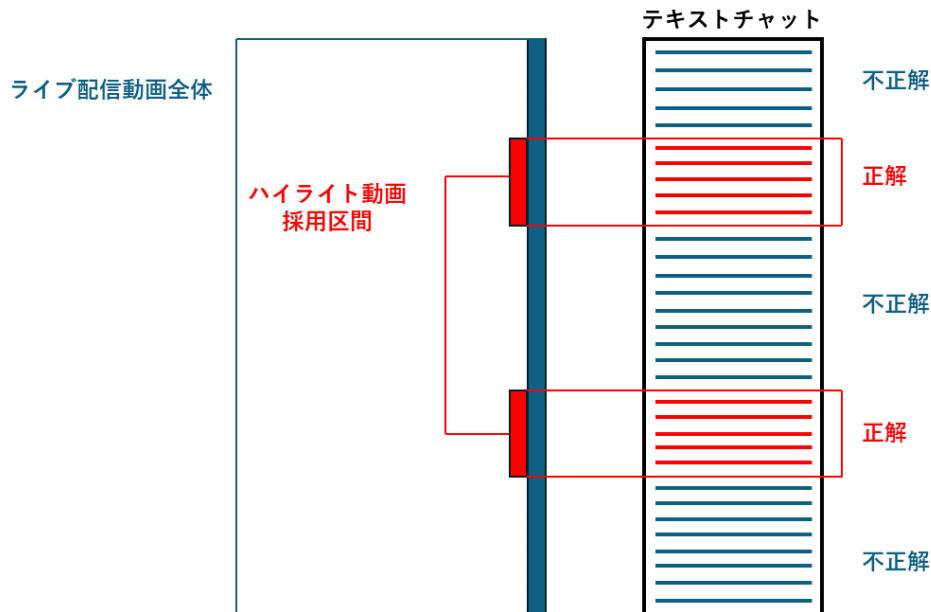


図 3.2: アノテーションの方法

区間ごとにキャプションがついており、これを利用する。図 3.2 の赤軸の部分において、ハイライト動画のキャプションを元にハイライト部分を判断し、そのハイライト部分が始まってから書き込まれたテキストチャットを正解としてアノテーションを行った。また、ライブ配信内のハイライト部分に対して視聴者が書き込むテキストチャットは、入力の手間やタイムラグなどから、ハイライト部分が終了した後もしばらく書き込みが続くことが予想される。そのため、キャプションから判断したハイライト部分が終了してから 20 秒後までに書き込まれたテキストチャットを正解とした。

次に、ゲームプレイのライブ配信動画内で、主に正解データ内に多くみられたポジティブな表現の内、不正解データ内にもみられた、[w, 草, 笑, ナイス, 惜しい]を含む表現と、感嘆表現のテキストチャットを正解とした。その際、例えば「ないす, おいしい」などの意味は同じだが表記が違う表現についても正解とし、これらに該当しないデータを不正解としてアノテーションを行った。

3.2.2 WRIME データセットの利用

別のタスクに利用した既存データセットを、目的のタスクのデータセットとして疑似的に利用した研究 [15] がある。この研究では、アノテーションを行うことなく、ソーシャルメディア上のテキストの感情特性を表すことができる可能性を示している。そこで本研究では、不正解データに WRIME データセット [16] を利用して、不正解データセットの疑似的な作成を試みる。WRIME データセットは、Github で公開されている日本語の感情分析用のデータセットで、SNS に投稿された文章と、その文章が表す 8 種の感情表現（喜び、悲しみ、期待、怒り、驚き、怖れ、嫌悪、信頼）と 4 段階の強度（無、弱、中、強）のラベルがついている。SNS に投稿されたテキストはライブ配信中に書き込まれるテキストチャットと文体や表現が類似していると考えられるため、疑似的な不正解データとして WRIME データセットを選択した。WRIME データセットのアノテーションは、SNS のテキストの書き手と、読み手 3 名が行っているが、読み手による客観的なアノテーションデータの方が感情強度推定の精度が高かったことから、本研究では読み手 3 名の付けたラベルの平均を用いる。疑似的な不正解データとして、ハイライト部分に対して書き込まれることが少ないと想定される「悲しみ、怒り、恐れ、嫌悪」の感情いずれかの感情強度が中以上のテキストと、すべての感情表現の強度が弱以下のニュートラルな感情のテキストを利用する。WRIME データセットを不正解データとして利用して学習を行ったモデルの精度評価は、YouTube から取得したアノテーション済みデータのみを用いる場合と、WRIME データセットを不正解データとして用いた場合の両方を用いて行う。

3.2.3 データセットの前処理

YouTube から取得したテキストチャットには、ライブ配信を行っているアカウントに対して、月額料金を支払い自身のアカウントを登録することで、使用することのできるスタンプがある。スタンプには様々な種類があり、スタンプも重要な表現である可能性はあるが、本研究では扱わない。また、「こんにちは（スタンプ）よろしく」の様に文章内にスタンプが使用されている場合は、「こんにちはよろしく」とし

て、スタンプ以外の文章はそのまま利用する。また、宣伝目的で URL を含む文章については、URL 以外の文章についてもライブ配信の内容に対して書き込まれたテキストチャットではないことが想定されるため除外する。次に、連続文字列は2文字までに制限をしたうえで、完全に同一の表現は一つを残して除外した。その際、ラベル1とラベル0のテキストチャットについて同一の表現があった場合は、ラベル1のテキストチャットを残した。最後に、日本語 BERT の Tokenizer を用いて、テキストチャットの埋め込み表現を獲得する。ファインチューニングの際のバッチ処理では、バッチ内のすべてのサンプルのトークン数が一致している必要があるため、トークン数が BERT の最大入力受付数である 512 になるようパディングを行った。バッチ処理のためのパディングであるため、BERT の Attention 機構によってモデルはパディング部分の情報を参照しない。

WRIME データセットについては、1つのデータ内に複数の文章が含まれているデータがあるため、先述の前処理に加えて、文章の最初に現れる「!’, ‘?’’, ‘。’, ‘、’」の四種類の記号まで取得し、それ以降のテキストはデータから削除した。また、正解データと不正解データの数に不均衡がみられる場合は、多い方のデータのダウンサンプリングを行うことによって比率が 1:1 になるように調整した。学習には、前処理が完了したデータを、8割を学習用データ、1割をテスト用データ、1割を検証データとしてランダムに分割し使用する。

3.3 モデルの概要

本研究では、2.3.1 節で述べた BERT をファインチューニングして、入力されたテキストチャットの2値分類を行うモデルを作成した。また、本研究では、BERT 内の層のパラメータの更新は行わず、追加した Linear 層及び、CNN の畳み込み層のパラメータを更新する。

3.3.1 BERT+Linear

BERT+Linear モデルの概形を図 3.3 に示す。図における、[CLS], E, [SEP], Pad は BERT への入力であり、[CLS], [SEP] は特殊トークンであり、文章の始めと終

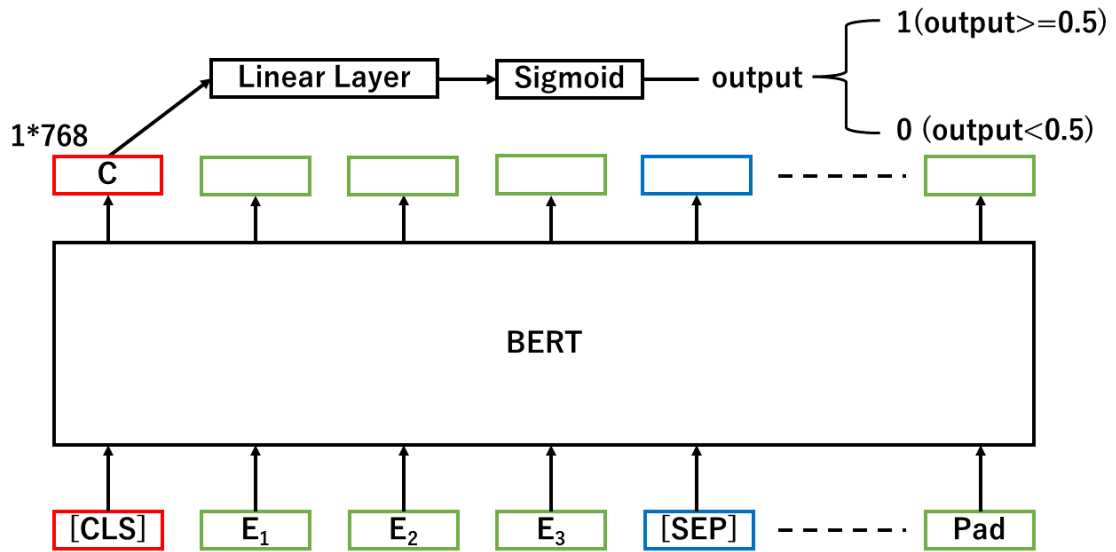


図 3.3: BERT+Linear モデルの構造

わりに挿入される。Eは文章内の各トークンの埋め込み表現である。また、Padはバッチ処理のために行われたパディング部分を示している。BERTへの入力の[CLS]に対応する、出力層におけるCには、入力文章全体の特徴量が含まれており、テキスト分類はこのCトークンに含まれる768次元の特徴量を持つベクトルを用いて行う。ベクトルを 768×1 のサイズのLinear層に入力し、1次元のスカラー値を取得する。その値をSigmoid関数に入力し0~1のoutputを出力する。最後に、outputの値を、0.5以上であれば1、0.5未満であれば0を出力することで、入力されたテキストチャットの2値分類を行うモデルを作成した。

3.3.2 BERT+pooling

BERTの出力層における図3.3のC以外のトークンも各入力トークンの情報を持っており、実際にSentence-BERTではBERTの出力をPoolingすることで、C以外のトークンの情報を利用している。そこで、本研究においてもPooling層を追加することで、C以外のトークンが持つ情報を使用するBERT+Poolingモデルを作成した。モデルの構造を以下の図3.4に示す。図3.4における入力、Linear Layer及びoutputの扱いは図3.3と同一である。また、図3.4におけるnはPad以外の入力トークン数

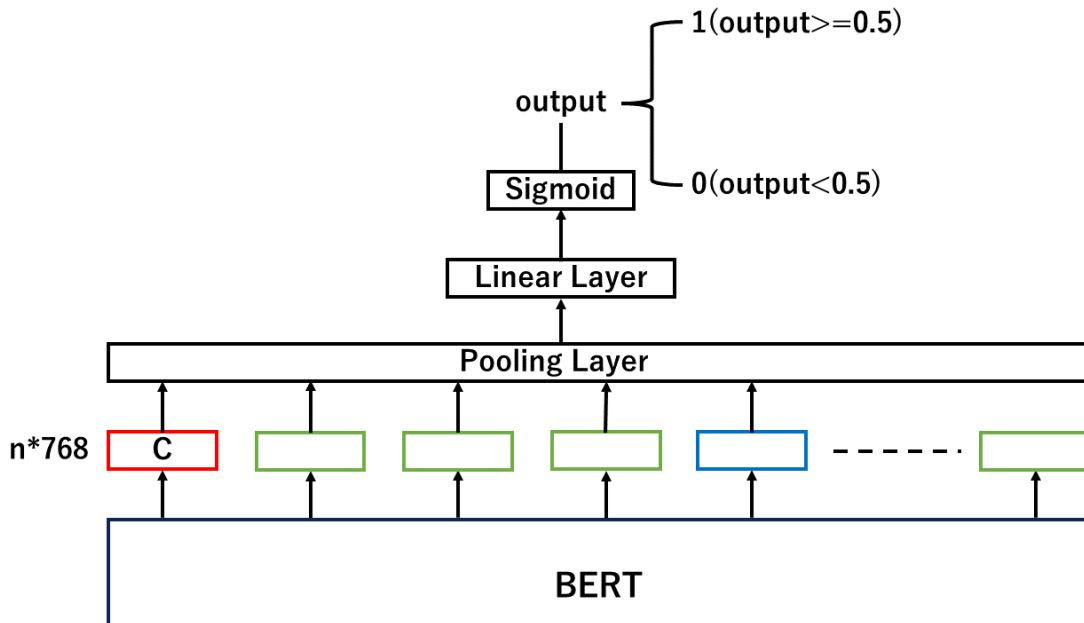


図 3.4: BERT+Pooling モデルの構造

を示しており、同図においては5となる。BERT の出力の内、Pad に該当しない部分のトークンの特徴量に対して Average Pooling を行い、768 次元の特徴量を算出する。その後の処理は 3.3.1 節と同一である。

3.3.3 BERT+CNN

BERT の出力における図 3.3 中 C 以外のトークンの特徴量を用いる手法として、BERT の出力層に CNN を追加し、各トークンの特徴量の畳み込みを行う手法が提案されている [12]。図 3.5 に本研究で使用した BERT+CNN モデルの構造を示す。図中 CNN Layer におけるハイパーパラメータは、kernel_size=2, out_channel=150 とした。BERT の出力のサイズバッチサイズ \times 512 \times 768 に対して畳み込みを行い、バッチサイズ \times 150 \times 767 の出力を得る。その後、出力の 2 次元目に対して Max Pooling を実施し、バッチサイズ \times 767 のサイズのベクトルを得る。その後、サイズ 767 \times 1 の Linear 層へ入力し、バッチサイズ \times 1 の出力を得る。その後の処理は 3.3.2 節と同一である。

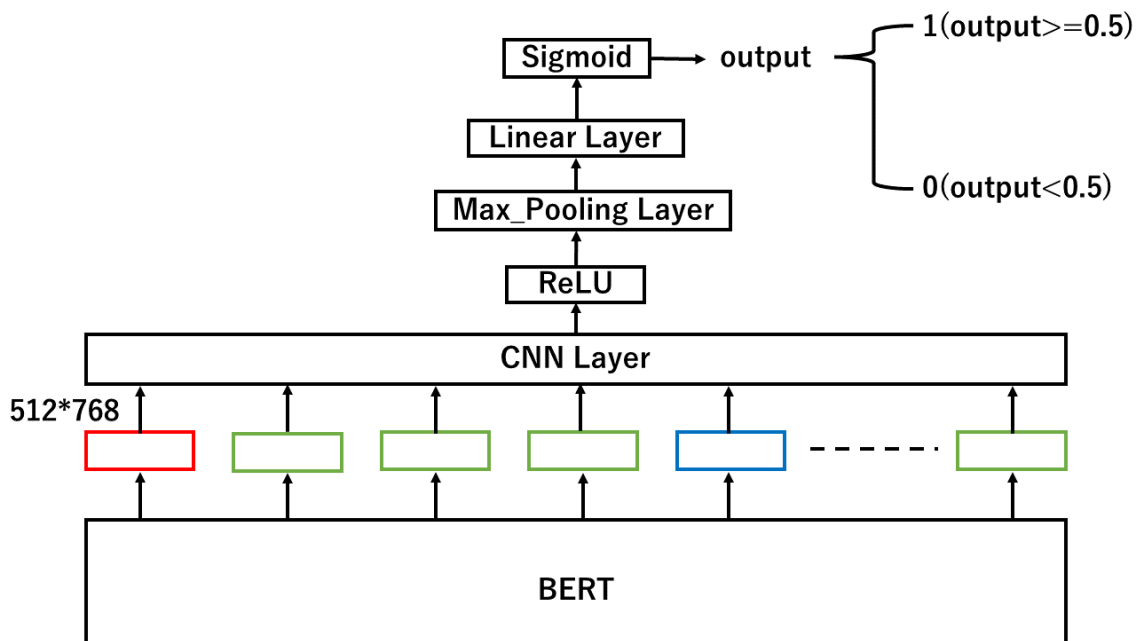


図 3.5: BERT+CNN モデルの構造

3.3.4 Sentence-BERT

ハイライト部分に対する視聴者のリアクションは、人によって大きな差異は無いと仮定し、Sentence-BERT を用いてテキストチャット同士の類似度を計算し、テキストチャットの表現が似ている箇所をハイライト部分と推定するモデルを作成する。具体的には、ライブ配信中に書き込まれたテキストチャットを20個ずつに分割し、すべてのテキストチャット間において類似度を計算し、その平均値が大きい順に20個のテキストチャットが書き込まれた区間とその前20秒の区間をハイライト動画に含めていく。他のモデル同様、10秒を1区間として、テキストチャットの類似度が高い区間を算出しない理由は、テキストチャットの書き込みが少なかった10秒間に類似度の高い文章が2つのみ書き込まれていた場合に、その区間をハイライト動画に含めてしまうことを避けるためである。また、Sentence-BERT については学習を行わず、テキストチャット同士の類似度の算出のみを行う。そのため、3.4節では、出力結果をもとに作成したハイライト動画についての考察を述べる。

3.4 学習の結果

本章で述べたモデル, およびデータセットを用いて各モデルの学習を行った. YouTubeから取得したテキストチャットのみを用いての学習では, 6594件のテキストチャットデータを, 不正解データにWRIMEデータセットを利用した学習では, 6190件のデータを用いて学習を行った. また, WRIMEデータセットを用いて学習を行った場合のテストには, 不正解データ, 正解データともにYouTubeから取得したテキストチャットを用いている. 学習は50 epoch行い, 各epochごとに損失と正解率を検証用データを利用して算出した. また, テストには, 50 epoch内で検証データに対する損失が最低値のモデルを利用し, 評価には以下の4つの指標を用いて式3.1~3.3で算出されるものを用いる.

- TP: 正解データに対してモデルが正解と判定した数
- FP: 不正解データに対してモデルが正解と判定した数
- TN: 不正解データに対してモデルが不正解と判定した数
- FN: 正解データに対してモデルが不正解と判定した数

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP} \quad (3.1)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3.2)$$

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (3.3)$$

YouTubeから取得したテキストチャットのみを用いて学習を行った各モデルにおける結果を表3.2に示す. 表におけるRecallの値から, 正解データの内, 正解と判定できたデータが7割程度だった. これは, 正解データと不正解データの分別がまだ不完全であることが原因だと思われるが, 前処理を追加したことによって, モデルが正解とだけ判定する問題を解決できている. また, どのモデルの精度でも80%を上回っているが, モデルの違いによる精度の向上はない.

次に, WRIMEデータセットを不正解データとして疑似的に利用し学習を行った結

表 3.2: YouTube から取得したデータセットのみを用いて学習したモデルの精度

モデル名	Accuracy	Recall	Specificity
BERT+Linear	0.82	0.71	0.93
BERT+Pooling	0.80	0.75	0.84
BERT+CNN	0.81	0.68	0.94

表 3.3: WRIME データセットを用いて学習したモデルの精度

モデル名	データセット	Accuracy	Recall	Specificity
BERT+Linear	YouTube	0.59	0.91	0.28
	YouTube+WRIME	0.92	0.93	0.91

果について述べる。表 3.2 より、モデルの違いによる精度の変化がみられなかったことから、BERT+Linear モデルの結果についてのみ表 3.3 に示す。表 3.3 より WRIME データセットのみを用いて、実際の正解データと不正解データを分類することはできなかったことが確認できる。しかし、WRIME データセットと正解データのは 92% で分類できていることから、それぞれのデータセットは異なる特徴を持っていると考えられる。また Specificity, と Recall をみると、正解データの分類精度を落とすことなく、不正解データについても 28.3% 程度分類できている。これは、WRIME データセットの利用によって不正解データのうち、WRIME データセットと似た特徴をもつデータが 28.3% あったことを示している。つまり、WRIME データセットの使用によって、YouTube から取得したテキストチャットのみでは、不正解データとして判定できなかった不正解データを、正解データの分類精度に影響を与えずに判定できる可能性を示している。また、今回は感情分析用のデータセットである WRIME データセットを利用したが、より不正解データに似た文法や、表現のデータセットを用いることで、不正解データの疑似的な代用が可能であると考えられる。

また、日本語 Sentence-BERT の利用によって出力されたハイライト動画を確認したところ、ライブ配信開始直後や終了間際のシーン、配信を行っているユーザーが離席する際や戻ってくるシーンなど、ハイライトとは無関係と考えられるシーンがハイライト動画に含まれてしまっていた。視聴者が書き込んだテキストチャットの類似性を用いたハイライト動画生成では、ライブ配信動画内のハイライト部分でない区間に起きたイベントに対して、視聴者が類似度の高い文章を書き込んでいた

場合に、その区間をハイライト動画に含めてしまうという問題や、例えば面白い出来事に対する表現である、「w, 草, 笑」はインターネット上で似た表現として使われるが、これらの類似度が高く算出されないという問題がある。そのため、テキストチャットの類似度を用いてハイライト動画を生成するためには、インターネット上で用いられる表現の類似度を学習させることや、ハイライト部分でない区間に多くの視聴者が似た反応をしている部分のテキストチャットの特徴量が算出された場合には、テキストチャット間の類似度が高い場合でも、ハイライト動画に含めないなどの処理が必要である。しかし、Sentence-BERT 自体の学習には膨大なデータセットとリソースが必要であるため、本研究では行わない。

以上の内容を踏まえて、ハイライト動画の生成には、YouTube から取得したテキストチャットのみを用いて学習を行った BERT+Linear モデルを用いる。

第4章 評価実験

本章では、第3章で述べた手法を用いてハイライト動画を生成し、被験者から得た評価によって提案手法の有用性の評価を行う。

4.1 ハイライト動画生成

学習用に用いたライブ配信動画 A の配信者の葛葉が VALORANT をプレイしている別のライブ配信動画 B を対象とする。表 4.1 に使用したデータを示す。B の長さは、6 時間 6 分 46 秒で、取得したテキストチャットの総数は 31753 件である。取得したテキストチャットに対して、3.2.3 節で述べた前処理のうち、URL を含むテキストチャットの削除、連続文字列の制限、スタンプの削除の前処理を行い、3.4 節で選択したモデルを用いて、ハイライト動画の生成を行った。また、生成するハイライト動画の長さが、葛葉の行ったライブ配信のハイライト動画のみを投稿しているアカウントの内、最もチャンネル登録者数が多かった「葛葉切り抜き ch」が B を元に作成したハイライト動画に最も近づくように区間数を決定する。

4.2 被験者による評価実験

提案手法を用いて生成したハイライト動画を被験者に評価してもらい、提案手法の有用性を確認する。被験者には提案手法により生成したハイライト動画の評価のために、その動画を含む以下 4 つの動画を以下の順番で視聴してもらった。

1. ライブ配信動画 B の冒頭 40 分
2. 葛葉切り抜き ch が作成したハイライト動画の内 B の冒頭 40 分に含まれる区間 (7 分 58 秒)
3. B の冒頭 40 分を元に提案手法により生成したハイライト動画 (8 分)

表 4.1: 評価用に用いたデータ

アカウント名	使用したデータ	動画 URL
Kuzuha Cannel	ライブ配信動画 B テキストチャット	https://www.youtube.com/watch?v=1YJE1CBf_2o
葛葉切り抜き ch	ハイライト動画 (B を元に作成)	https://www.youtube.com/watch?v=KLeU0fCkIvY

4. B の冒頭 40 分を元に区間をランダムに選択し生成したハイライト動画 (8 分) 被験者の負担を考え、すべての動画を視聴した際にかかる時間が 1 時間程度になるように、ハイライト動画生成の対象を B の冒頭 40 分とした。また、提案手法を用いて作成したハイライト動画の有用性の確認のために、B の他に、葛葉切り抜き ch が B をもとに作成したハイライト動画と、区間をランダムに選択し自動生成したハイライト動画を視聴してもらった。区間をランダムに選択する際には、区間の選択以外に提案手法との差が出ないように、10 秒を 1 区間としてランダムに選択された区間と、その前の 2 区間の合計 3 区間をハイライト動画に含め、区間が重複した場合は、提案手法と同様に区間の併合を行った。

各動画を視聴してもらった被験者は、葛葉の行ったライブ配信を視聴したことがあり、VALORANT のゲーム内容を知っている人に限定した。本研究では、多くの人が良いと感じるハイライト動画の作成を目的としているが、どのようなハイライト動画が良いかは、人によって差がある。そこで被験者には、ハイライト動画の質に関する特定の質問を設けず、提案手法を用いて自動生成したハイライト動画、葛葉切り抜き ch が作成したハイライト動画、区間をランダムに選択して自動的に生成したハイライト動画についての評価をそれぞれ最も悪ければ 0、最も良ければ 10 とし、それぞれ 11 段階 (0~10) で回答してもらった。また、提案手法により自動生成したハイライト動画についての評価を、自由記述回答で任意で行ってもらった。結果を表 4.2 に、それより作成した箱ひげ図を図 4.1 に示す。表 4.2 において、提案手法を用いて作成したハイライト動画の評価を「提案手法」の項目に、葛葉切り抜き ch が作成したハイライト動画の評価を「切り抜き ch」の項目に、ランダムに区間を選択して自動生成したハイライト動画の評価を「ランダム」の項目に記載している。

まず、図 4.1 より、提案手法を用いて自動生成したハイライト動画の評価につい

て、最低値が4、中央値が7である。加えて、平均値が6を超えており、1や2などの極端に低い評価がなく、全体的には良いハイライト動画という評価が得られた。また、自由記述の多くに、ライブ配信におけるハイライト部分をハイライト動画に含められていたという趣旨の記述があったことから、提案手法により、ライブ配信内のハイライト部分を抽出することが出来ているとみられる。次に、葛葉切り抜きchが作成したハイライト動画の評価については、すべての被験者が7以上の評価をしており、中央値や平均値からも被験者全体から高い評価を得られている。また、提案手法を用いて作成したハイライト動画を、葛葉切り抜きchの作成したハイライト動画よりも高く評価した被験者がいなかったことから、提案手法では葛葉切り抜きchの作成したハイライト動画よりも被験者にとって良いハイライト動画を生成できなかった。これらのことから、提案手法を用いて生成したハイライト動画は被験者から、人気のハイライト動画作成者が作成したものには及ばないが、良い評価を得られており、提案手法によってライブ配信動画の視聴を必要とせず、被験者が良いと感じるハイライト動画の生成を自動的に行えていると考えられる。また、ランダムに区間を選択して自動生成したハイライト動画の評価の内、提案手法を用いて自動生成したハイライト動画の評価を上回ったものが無いことや、「どちらの切り抜きも元動画の面白かったところを抑えられていたのが良い点」、「公開されているハイライト動画と切り抜いてる部分が同じで、面白いと感じる箇所が切り抜かれており、自動生成したハイライト動画としてはすごいと感じた」などの自由記述の評価から、提案手法によってライブ配信内のハイライト部分の推定が行えていると考えられる。そのため、ハイライト動画を作成する際に提案手法を用いることで、ライブ配信の内容をすべて視聴せず、ハイライト部分を確認できるため、効率的にハイライト動画を作成することができる可能性がある。最後に、自由記述における提案手法を用いて出力したハイライト動画への評価で、「前後の脈絡が入ってきにくい印象があった」や「冒頭部分など話がぶつ切りにされている箇所がやや気になった」など、葛葉切り抜きchが作成したハイライト動画に比べてハイライト部分の理解のために必要な区間が足りていないことによる否定的な評価が多かった。提案手法では、ハイライト部分と推定した区間とその前の2区間の合計3区間をハイライト動画に含め

ているが、ハイライト部分によってハイライト部分の前の区間をどの程度含めるか決定することで、より良いハイライト動画の自動生成を実現できると考える。

表 4.2: 被験者実験結果

被験者	提案手法	切り抜き ch	ランダム	自由記述
A	7	8	4	どちらの切り抜きも元動画の面白かったところを抑えられていたのが良い点。 元の動画をうまくまとめられていたように思う。
B	8	10	2	研究で出力した動画は40分の動画の要点を8分で見られるため退屈な時間が少なく視聴しやすい。 実際に公開されているハイライト動画と比較すると、切り抜いている箇所は大体同じだが切り抜き開始、終了点の差で前後の脈絡が入ってきにくい印象があった。
C	6	8	1	どちらの動画も切り抜き箇所については問題なく楽しめた。

被験者	提案手法	切り抜き ch	ランダム	自由記述
D	7	10	4	<p>公開されているハイライト動画と切り抜いてる部分と同じで、面白いと感じる箇所が切り抜かれており、自動生成したハイライト動画としてはすごいと感じた。</p> <p>自動生成したハイライト動画は、会話の最初の大事な部分が抜けている箇所があり、理解できなかった部分や、どういった経緯でこの会話になったかがわかりにくいところがあった。</p>
E	7	8	5	<p>ハイライトされた部分はほとんど同じであり、同時に盛り上がっていたシーンだった。</p> <p>また、いわゆるオチのシーンはハイライトされていたが、話の前提部分が途中から始まっていることが多く感じた。</p>
F	7	8	1	<p>元配信をみて面白いと思った場面がハイライト動画にも入っていてよかった。</p> <p>話のオチが切り抜かれていない（切り抜き区間が終わるのが早い）ところがあるように感じた。</p>

被験者	提案手法	切り抜き ch	ランダム	自由記述
G	4	8	2	クリップを集めたもののような印象を受けた。 公開されているハイライト動画と比べて会話などの流れが掴みづらかったが比較対象より網羅してる範囲が広いのは良い。
H	8	10	2	自動生成したハイライト動画と、既存のハイライト動画の両方とも、面白いシーンをハイライト動画に含んでいた。 面白いシーンが何故起こったのかの説明的な部分が含まれていないところがあったので、そこを改善できればより良いハイライト動画の生成ができると感じた。

被験者	提案手法	切り抜き ch	ランダム	自由記述
I	5	10	3	<p>公開されているハイライト動画は面白いと感じる場面に至るまでをテンポよくまとめられているのに対し、自動生成したハイライト動画は何も会話がないう部分や面白いと感じる場面とは関係のない会話が入ってしまったり、もしくは関係のある部分が一部損なわれていることで盛り上がり切り抜き動画よりも下がっているように感じた。</p> <p>切り抜いた部分に関してはどれも配信の中で面白い場面ではあった。</p>
J	4	8	1	<p>会話のブツ切りや、話の流れが分からなくなるカットの仕方があった。</p> <p>ハイライトとして入れた方がよい場面がいくつか無かった。</p>
K	6	7	3	<p>提案手法で出力した動画、切り抜き動画両者とも面白いところを切り抜いておりあまり差異はなかったが、前者の冒頭部分など話がぶつ切りにされている箇所がやや気になった。</p>

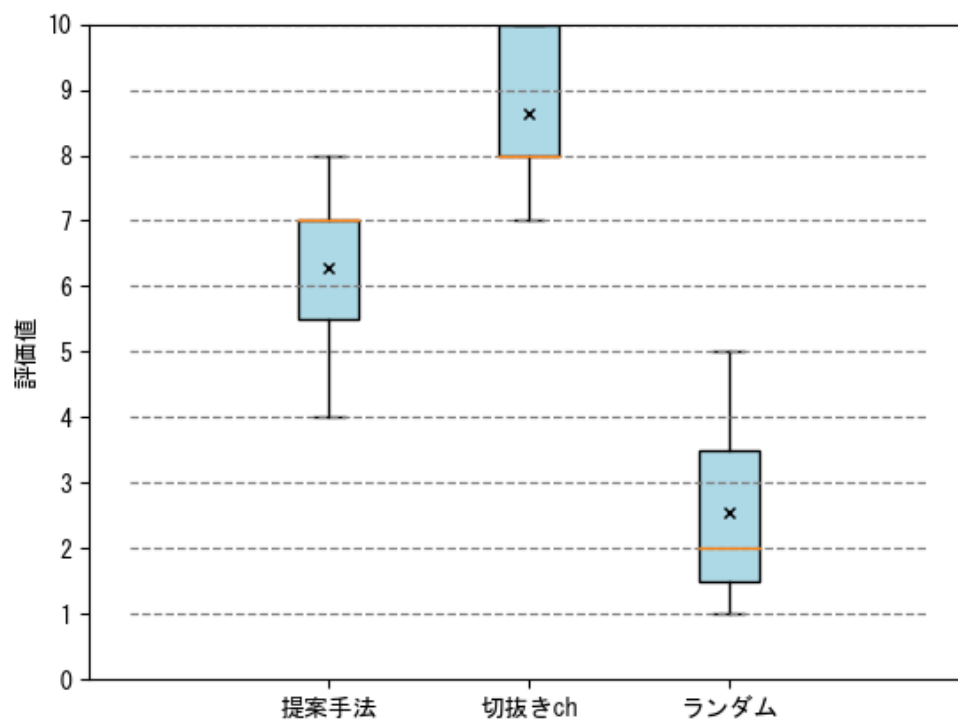


図 4.1: 被験者実験結果より作成した箱ひげ図

第5章 むすび

本研究では、ライブ配信中に視聴者が書き込んだテキストチャットを利用してライブ配信のアーカイブ動画からハイライト動画を自動的に生成した。まず同一のライブ配信動画から作成されたハイライト動画を元にアノテーションを行い機械学習モデルをファインチューニングした。また、ファインチューニングしたモデルの出力結果を集計し、視聴者が書き込んだテキストチャットのうち、ハイライト動画として採用される区間に対して書き込まれたと判定したテキストチャットが多い区間順にハイライト動画として採用した。不正解データに含まれる正解データと同じ内容のデータによってファインチューニングができず、モデルが入力されたテキストチャットをすべて正解と判定する問題を、特に正解データに多く見られた特定の単語や表現を含む不正解データを正解として扱うことで解決した。また、WRIME データセットの使用によって、正解データの分類精度を落とすことなく、不正解データの分類精度を向上できる可能性を示した。最後に、オリジナルのライブ配信を視聴したことがある被験者を対象に、自動的に生成したハイライト動画を視聴してもらった。提案手法により、本研究の目的である、多くの人が良いと感じるハイライト動画の生成を行えたことを確認した。また、提案手法において、推定したハイライト部分ごとに直前の区間をどの程度ハイライト動画に含めるか動的に判断することで、視聴者がより良いと感じるハイライト動画の生成が行えることが示唆された。

今後の課題としては、ゲームプレイ以外のジャンルのライブ配信動画に対する本研究の提案手法の有用性の確認や、提案手法の定量的な評価を行うことが挙げられる。また、不正解データに存在する正解データと同様の意味を持つテキストチャットの影響を低減するために、テキストチャットがハイライト部分に対して書き込まれたものかどうかのアノテーションを人手で行うことや、WRIME データセット以外に不正解データの代用として利用できる、新たなデータセットによる疑似的な不

正解データの利用が挙げられる。最後に、本研究で提案した手法の評価方法として、実際に人気のハイライト動画を作成している人の協力を得て、ライブ配信動画におけるハイライト部分の重みづけを行い、ライブ配信における重要なハイライト部分をどの程度ハイライト動画に含めることができていたか確認することで、システムの評価を定量的に行うことが挙げられる。

謝辞

本研究においてご指導いただいた名古屋工業大学，舟橋健司 准教授，伊藤宏隆 助教に心から感謝致します。また，本研究で提案したシステムの評価に協力して頂いたすべての方に感謝致します。また，本研究を進めるにあたり多くの助言とご協力を頂いた，名古屋工業大学 舟橋研究室の皆様に感謝致します。本研究は，JSPS 科研費 JP19K03028 の助成を受けたものです。

参考文献

- [1] 保高隆之, 阿曾田悦子 “コロナ禍はテレビと動画の利用者にどんな影響を与えたか 「コロナ時代のテレビの価値」調査の結果から”, 放送研究と調査, Vol.71, no.10, pp.2-33, 2021.
- [2] Think with Google, “YouTube をマーケティングのヒントに —— トrendが生まれ、マルチフォーマットなどで生活に定着進む”, <https://www.thinkwithgoogle.com/intl/ja-jp/marketing-strategies/video/yt-trendsreport2022/>. (Accessed on 01/06/2024).
- [3] Wei-Ta Chu, Yung-Chieh Chou, “On broadcasted game video analysis: event detection, highlight detection, and highlight forecast,” Multimedia Tools and Applications, Vol.76, pp.9735-975, 2017.
- [4] 小城凱, 伊東栄典, “動画のハイライト部分自動抽出に向けた検討”, 情報科学技術フォーラム講演論文集, 22 卷 (第 2 分冊), pp.473-474, 2023.
- [5] 植野天翔, 平井辰典, “BERT を用いてライブ配信のチャットから視聴者の好みに合ったハイライト動画を生成するシステム”, 第 85 回全国大会講演論文集, 2023, pp.763-764, 2023.
- [6] “MeCab: Yet Another Part-of-Speech and Morphological Analyzer,” <https://taku910.github.io/mecab/>. (Accessed on 01/06/2024).
- [7] D.P. Kingma, J.L. Ba, “Adam: A Method for Stochastic Optimization,” <https://arxiv.org/abs/1412.6980><https://arxiv.org/abs/1412.6980>, 2015. (Accessed on 01/06/2024).

- [8] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *Association for Computational Linguistics*, Vol.1, (Long and Short Papers), pp.4171-4186, 2019.
- [9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, “Attention Is All You Need,” <https://arxiv.org/abs/1412.6980>, 2017. (Accessed on 01/06/2024).
- [10] D. Bahdanau, K. Cho, Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate,” <https://arxiv.org/abs/1409.0473>, 2016.
- [11] “Hugging Face,” <https://huggingface.co/cl-tohoku/bert-base-japanese-whole-word-masking>. (Accessed on 01/06/2024).
- [12] K. Kaur, P. Kaur, “BERT-CNN: Improving BERT for Requirements Classification using CNN,” *Procedia Computer Science*, Vol.218, pp.2604-2611, 2023.
- [13] N. Reimers, I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp.3982-3992, 2019.
- [14] “Hugging Face,” <https://huggingface.co/sonoisa/sentence-bert-base-ja-mean-tokens-v2>. (Accessed on 01/06/2024).
- [15] R. Yamanishi, H. Takemoto, Y. Nishihara, M. Yoshida, T. Ohsuga, K. Oyama, “Applying Existing Dataset as a Pseudo Corpus for Sentiment Representation on Social Media,” *Procedia Computer Science*, Vol.207, pp.335-342, 2022.
- [16] T. Kajiwara, C. Chu, N. Takemura, Y. Nakashima, H. Nagahara, “WRIME: A New Dataset for Emotional Intensity Estimation with Subjective and Objective Annotations,” In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp.2095-2104, 2021.

発表論文リスト

投稿論文

1. T. Kamimura, H. Itoh, K. Funahashi, “Analysis of the impact of COVID-19 on student life based on the results of a survey,” 2022 Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems (SCIS&ISIS), T-3-F-1, 2022.