ファインチューニングしたBERTとGPT-4oを用いた相談・助言形式動画の自動切り抜き

岡崎 光栄[1,a)]　北原 鉄朗[1,b)]

概要：本研究では, ファインチューニングした日本語文脈理解モデル TohokuBERT と GPT-4o を用いて, 視聴者からの質問や相談を出演者が読み上げ, 同じ出演者が回答や助言を行う単独出演動画から「相談」と「助言」を自動的に切り抜き, 要約するシステムを試作した.YouTube などの動画共有プラットフォームで需要が高まる切り抜き動画の自動生成を目指し, 音声認識技術で動画から音声を書き起こし, TohokuBERT で各文を「相談」,「助言」,「その他」に分類する. その後,「相談」と「助言」が連続するグループを特定し, GPT-4o を用いて文脈の一貫性を評価・修正するプロセスを構築した. 結果, TohokuBERT は一定の分類精度を示したものの, 音声認識の誤りにより一部の文が誤分類されるケースが確認された. また, GPT-4o による文脈評価では, 一貫性の欠如する文が残留する課題が明らかとなった.

1. はじめに

近年, YouTube をはじめとする動画共有サービスの普及により, 長時間の動画から視聴者が求める特定部分のみを抜粋・再構成する切り抜き動画が注目を集めている. しかし, 動画編集作業では編集ソフトを用いて開始点や終了点を正確に指定する高度な操作技術が求められ, 動画を繰り返し確認するなどの多大な時間的コストも発生する. 従って, 作業を効率化する自動化手法が求められる. 本研究では, 視聴者からよせられた質問や相談を出演者が読み上げ, 同じ出演者が回答や助言を行う, 単独出演の動画を対象とし, 抜粋・再構成した切り抜き動画を自動生成することを目的とする.

既存の研究では, 動画の文字起こしや字幕データに基づくテキスト要約手法が数多く提案されている [1], [2], [3], [4], [5]. これらはテキスト生成や要約精度の向上に特化しており, 映像要素との統合や文脈を考慮した映像要約には必ずしも対応していない. 一方で, 動画全体を対象とし, 映像や音声などの情報を解析して要約・編集を行う手法も存在する [6], [7], [8], [9], [10]. これらの研究は, ユーザのクエリや映像的特徴を手がかりに重要部分を抽出し, 必要な場面のみを再構成する点で有用である. しかし, その多くは与えられた指示や映像情報に基づく要約を主としており,「相談」から「助言」へと展開する特定の文脈を考慮した動画要約には対応していない. 言い換えれば, クエリ依存の整理や映像特徴量の分析に強みはあるものの, 特定の文脈を踏まえた要約は行われていない.

我々は以前, GPT-4 を用いて「質問」「回答」箇所を特定し, speech-to-text の latest_long モードよる書き起こしとの照合により切り抜き範囲を自動抽出するシステムを試作した [11]. 質問（相談）と回答（助言）のペアをいくつか抽出することはできたものの, ペアが正しく形成されなかったり, 回答（助言）内の冗長な部分（特に話が逸れる部分）は, そのまま残されるなどの問題を抱えていた.

本稿では, この 2 つの問題を解決する切り抜き動画生成システムを試作する. 前者の問題に対しては, 日本語文脈理解モデルである TohokuBERT をファインチューニングし,「相談」と「助言」を高精度に分類できるモデルを構築する. 後者に対しては, 相談と助言のペアが形成された後, 助言に含まれる各文が相談文と文脈的に整合するかどうかを GPT-4o で判定し, 文脈的に整合する文のみを残すようにする. これらを通じて長時間動画から意味的一貫性を維持した要約動画の自動生成を実現する.

2. 提案システム

本システムは, 長時間の動画から重要部分を切り抜いた要約動画を自動で生成する. 本システムが対象とするのは, 1 人の人が, 視聴者から寄せられた質問や相談ごとを読み上げ, それに対して自身の考えを述べたり助言したりする, というものが繰り返される動画である. ここでは, 前者を「相談」, 後者を「助言」と統一的に呼ぶ. 本システム

[1] 日本大学文理学部情報科学科
[a)] okazaki@kthrlab.jp
[b)] kitahara@kthrlab.jp

図 1 本システムの処理の流れ

では, 各発話が相談か助言かをファインチューニングした TohokuBERT で分類し, さらに助言の各発話が相談発話と文脈的に整合しているかを GPT-4o を用いて判断する. 処理の流れを図 1 に示す.

2.1 音声抽出

動画から音声を抽出する. まず, yt_dlp で YouTube 動画をダウンロードし, Google drive に保存, そして, FFmpeg を使用して音声のみを取り出す. これにより, 後続の文字起こしに使用する音声データが得られる.

2.2 書き起こし

抽出した音声データに対して, Google Speech-to-Text API（STT）を用いて書き起こしを行う. この過程で以下のデータが得られる.

- 単語単位のタイムスタンプ付きデータ
  各単語に開始時間および終了時間が付与される.
- 句読点補完文データ
  文末に自動的に句読点（「。」や「?」）が挿入されるデータが出力される. ただし, 文単位のタイムスタンプは付与されない.

しかし, 提案システムでは文末表現（句読点）で区切って動画を切り抜く必要があるため, 単語単位のタイムスタンプのみでは正確な文単位の時間情報が得られない.

2.3 単語単位のタイムスタンプと文区切りの不一致問題

単語単位のタイムスタンプと句読点で区切られた文データを照合し, 文単位のタイムスタンプを割り出そうとした. しかし, 単語の区切り目と文末の句読点の区切り目が一致しない問題が発生した. STT が出力した単語のタイムスタンプデータと文データの句読点の位置が完全に揃わないため, 単語単位で文の開始・終了時間を特定することが難しい. この問題を解決するために, 単語単位の開始時間および終了時間を, 単語内のすべての文字に均一に割り当てる処理を行う.

2.4 文字単位のタイムスタンプ割り当て

単語単位で付与されたタイムスタンプを, 各単語内のすべての文字に対して同一のタイムスタンプを割り当てる. この処理により, 単語の開始時間と終了時間を用いて文字単位の時間情報が取得できる. これにより, 句読点で区切られた文のタイムスタンプを後続のステップで正確に抽出できるようになる.

(1) 句読点で文を分割
　書き起こしデータを句読点「。」および「?」で分割する.

(2) 文ごとの時間情報の取得
　分割された文に含まれる先頭の文字の開始時間を文の開始時間とし, 末尾の文字の終了時間を文の終了時間とする.

これにより, 正確な文単位のタイムスタンプが抽出可能となり, 文区切りの動画切り抜きに利用することができる.

2.5 分類タスクと文のグループ化

句読点区切りの文を基に, 動画内の発話を「相談」「助言」「その他」の 3 つのカテゴリに分類する. 分類された文は, 次のようにグループ化される.

- **「相談」グループ**
  視聴者から送られた質問や相談の読み上げに当たる文.
- **「助言」グループ**
  動画話者が相談に対して回答や助言を行う文.
- **「その他」グループ**
  雑談や動画の主旨と関係のない文.

ここで, 分類には高度な文脈理解を持つ自然言語処理モデルを用いるが, その詳細については第 3 節で述べる. 分類結果を基に, 「相談」と「助言」が連続する文のみをグループ化し, 動画の切り抜き対象とする.

2.6 文のグループ化

文脈理解モデルによって分類された「相談」および「助言」の文グループに対して, 同一ラベルが連続する文を一つのグループとしてまとめる過程である. 具体的には, 以下の手順でグループ化を行う.

(1) **ラベルに基づくフィルタリング**
　「その他」に分類された文を除外し, 残った「相談」と「助言」の文のみを対象とする.

(2) **連続する同一ラベルの文の統合**

同一ラベルが連続する文を一つのグループとしてまとめる. 例えば, 連続する「相談」文は一つのグループとして, 連続する「助言」文も一つのグループとして扱う. グループは同時に各文のタイムスタンプも保持する.

(3) **グループごとのタイムスタンプ調整**
各グループ内の文のタイムスタンプを基に, グループ全体の開始時間と終了時間を設定する. グループの最初と最後のタイムスタンプだけを使用すれば, グループ単位の切り抜きが可能である.

このグループ化により, 関連する文が一つのまとまりとして扱われ, 後続の文脈適合性評価や動画切り抜きの際に効率的な処理が可能となる. また, 文グループは, 動画切り抜き時に一括して処理されるため, 動画の一貫性と流れを維持することが容易になる.

2.7 GPT-4o を用いた文脈適合性判断

文グループに対して, GPT-4o を用いて文脈適合性を評価し, 不整合な文を識別・除去するプロセスである. 具体的には, 以下の手順で行う

(1) **文脈適合性の評価**
GPT-4o に対して, 各グループ内の文が「相談」から「助言」への自然な流れに沿っているかを判断するプロンプトを送信する. これにより, 文脈的に適切でない文や冗長な文を識別する.

(2) **不整合文の識別**
GPT-4o からのレスポンスを解析し, 文脈に適合していない文の番号を複数抽出する. レスポンスが「-1」の場合, 全ての文が文脈に適合していると判断し, 除去する文はない.

(3) **文の除去**
抽出された番号に基づき, 該当する文をグループの添字を元に除去する. これにより, 要約動画には文脈的に適切な部分のみが残る.

(4) **グループの間の添字の文を取り除く場合**
グループ内の文が一部取り除かれ, 連続性が失われた場合, グループを連続性が保たれるように分割する必要がある. 以下の表 1 に具体例を示す.

表 1 削除分割例 ($S_{1,\ldots,n}$ はラベルグループの各文)

初期状態	**ラベル文群**	**タイムスタンプ**
元グループ	S_1, S_2, S_3, S_4	00:00:10〜00:20:30
削除	S_1, S_2 **削除**, S_3, S_4	00:00:10〜00:20:30
削除分割後 (S_2 は 00:00:30〜00:05:00)		
グループ 1	S_1	00:00:10 〜 00:00:30
グループ 2	S_3, S_4	00:05:00 〜 00:20:30

この分割により, 動画切り抜き時に S_3 の開始時間と S_4 の終了時間を参照することで, S_2 の間の不要な部分を省略し, 連続性を保った要約動画を生成することが可能となる.

```
prompt = f"""
以下の#で囲まれた文は{label}文と判定された句読点区切りに添字が割り当てられた一連の文です。
一連の文中の{label}の文脈に沿っていない文があるならば、また、文として成り立っていない文があるなら、複数その文の番号を返してください。
全ての文が{label}の文脈に沿っている場合は-1 を出力してください。
複数番号を出力するときは必ず「,」区切りで出力してください。
#
{numbered_sentences}
#
"""
```

図 2 文脈整合性判断で使用するプロンプト

文脈整合性判断で実際に使用するプロンプトを図 2 に示す. label には各グループのラベル, numbered_sentences には, 各グループの各文に番号が割り当てたものを動的に入力する. また, #で囲っているのは, グループがプロンプトのどこであるかを理解しやすくする工夫である.

2.8 動画生成

評価を通過したグループに対応する動画部分をタイムスタンプに基づいて切り抜き, これらのクリップを結合して要約動画を生成するプロセスである. 具体的には, 以下の手順を踏む

(1) **タイムスタンプのマッピング**
グループに割り当てられた開始時間と終了時間を基に, 動画から該当部分を特定する. 具体的には, グループ全体の開始時間はグループ内の最初の文の開始時間, 終了時間はグループ内の最後の文の終了時間として設定する.

(2) **動画切り抜き**
特定された開始時間と終了時間に基づき, FFmpeg などの動画処理ツールを用いて動画から該当部分を切り抜く. この際, 動画のフォーマットを統一し, 後続の結合プロセスをスムーズに行うための準備を行う.

(3) **クリップの結合**
切り抜かれた複数のクリップを時系列順に結合し, 最終的な要約動画を形成する.

この修正により, タイムスタンプのマッピングが文単位ではなく, 同一ラベルの連続グループ全体に対して行われることが明確になる. これにより, 動画の一貫性と流れを維持しつつ, 効率的に要約動画を生成することが可能となる.

3. TohokuBERT をファインチューニングした分類モデル

本節では, 動画内の発話を「相談」, 「助言」, 「その他」に分類するために使用する BERT および日本語特化型 TohokuBERT について説明し, 教師データの作成からファインチューニングの手法と結果について述べる.

3.1 BERT モデルの概要

BERT（Bidirectional Encoder Representations from Transformers）は, Google が開発した自然言語処理（NLP）の事前学習済みモデルである. BERT は, Transformer の**自己注意機構（Self-Attention Mechanism）**を活用し, **双方向的な文脈理解**を実現することが特徴である. 従来の NLP モデルが単方向（左から右, または, 右から左）の文脈しか考慮できなかったのに対し, BERT は入力文の前後関係を同時に考慮し, より深い意味理解を可能にする.

3.2 TohokuBERT の特性

TohokuBERT は, **東北大学**の研究チームによって開発された日本語特化型 BERT モデルである. このモデルは, **大規模な日本語コーパス**（Wikipedia や青空文庫）で事前学習されている. 従って, 日本語は助詞や語順, 敬語表現が豊富である.

さらに, TohokuBERT をファインチューニングすることで, **「相談」「助言」「その他」**の分類精度を最大化できる.

3.3 正解データの作成

ファインチューニングのために必要な教師データを準備する. 切り抜き動画の文字起こしデータに対して, 「相談」「助言」「その他」 の 3 種類のラベルを手動で付与し, 基準データ (表 2 参照) として使用する.

表 2 基準データの内訳: 計 3778 文

カテゴリ	文数内訳
相談	1000 文
助言	1778 文
その他	1000 文

ラベル付けされたデータの例を表 3 に示す.

3.4 ファインチューニングの手法

TohokuBERT を 3 クラス分類タスク適用のため, 表 4,5,6 パラメータでファインチューニングを実施した.

表 4 入力データの形式

入力文	書き起こしテキストの 1 文
手動付与ラベル	「相談」(0), 「助言」(1), 「その他」(2)

表 3 相談文, 助言文, その他文の例

種類	文例	ラベル
相談	今の仕事の環境ではこれ以上成長しないと考えており海外挑戦したいと思っています. 海外行くメリットデメリット教えてください.	0
助言	メリットはデメリットは日本の辛く日本語通じる市日本の文化風習わかる知り合いもコネクションもあるので日本人って日本で働いてくれた方が楽なんすよ. ただその全然それもないし言葉も通じないとこに行くといろんなことが一時自分でやんなきゃいけないので割と楽しいですよ. なんでそれが楽しいと思える人は海外ないんじゃないかなと思います.	1
その他	画面の隅に何か映り込んでる気がするけどなんだろうこれ.	2

表 5 教師データの分割

訓練データ	80%（3022 文）
検証データ	20%（756 文）

表 6 学習設定

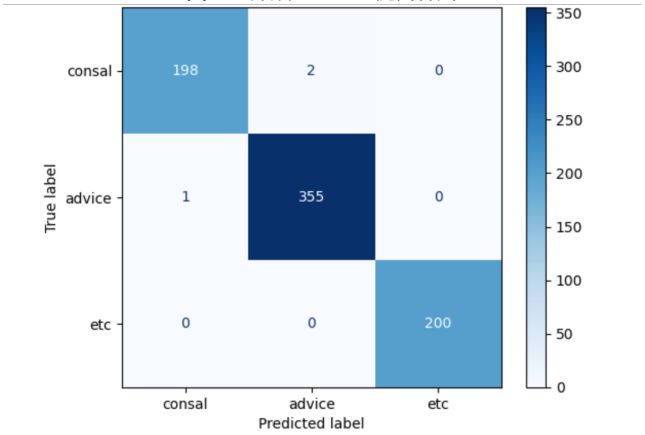
学習率	5×10^{-5}
バッチサイズ	16
エポック数	5
最適化手法	AdamW

訓練データを用いて学習し, **クロスエントロピー誤差**を損失関数として最適化を行い, 検証データで性能を評価した.

3.4.1 ファインチューニングの結果

学習済みモデルの評価には, 検証データに対する**混同行列**および **F1 スコア**を用いた. 混合行列を図 3, 精度（Precision）, 再現率（Recall）, F1 スコアを表 7 に示す. いずれのクラスも十分な精度で識別できていることがわかる. 誤分類の事例として, 「いいと思う?」のような疑問形の相談文が, 助言文と誤分類されるケースがあった. また, 文字起こし段階での誤認識や表記ゆれが, 分類精度に影響を与える可能性が示唆された.

図 3 分類タスクの混同行列



3.4.2 結果の考察

Precision と Recall はいずれも高く, 全体的に非常に高い分類精度が達成されている. F1 スコア もすべてのクラスで 0.99 以上を記録し, 誤分類が極めて少ないことが示さ

表 7　分類タスクの評価結果

ラベル	Precision	Recall	F1 スコア
相談 (0)	0.995	0.990	0.992
助言 (1)	0.994	0.997	0.995
その他 (2)	1.0	1.0	1.0

れる. ただ以下のような誤りも少なからず発生した.

- **誤分類の事例**
  「いいと思う?」のような相談文が, 発音の違いを判別できず, 助言文と誤分類されるケースが確認された.
- **表記ゆれの影響**
  文字起こし段階での誤認識や表記ゆれが, 分類精度に影響を与えることが示唆された.

4. 実際の実行例

視聴者からの質問や相談を出演者が読み上げ, 同じ出演者が回答や助言を行う単独出演動画に対して, システムの実行を実際に行う. 今回, YouTube 上で公開されている, ひろゆきの動画 3 つを用いて切り抜き動画を作成した.

(1)【ひろゆき】心の声より専門家に従ったほうがいいよの巻. なんかみながら. 2023/03/07 M21（前から 60 分) [12]
(2)「男女給与格差の原因は差別ではなく母親. 小諸のワインを呑みながら 2023/11/05 L23」（前から 60 分）[13]
(3)「独立より副業のほうが成功率が高い.Bière du Vexin を呑みながら 2023/10/14 S19」（60 分）[14]

これらはいずれも，YouTube Live のチャット欄に寄せられた質問を読み上げた上で，ひろゆきが自身の意見を述べる形式が含まれる動画である.

4.1 1 つ目の動画 [12]

実行結果として得られた時系列順上から 6 つのグループを抜粋し, 表 8 に示す.

動画は, 18 分 16.8 秒に要約された. また, 表 8 を観察すると, グループ 1 からグループ 2, グループ 3 からグループ 4 の「相談」と「助言」の文脈が一貫していないことが明らかになった. また, グループ 5 については, 相談文ではあるように見えるものの, 具体的な内容が不明確である.

しかし, 以下のような文脈が成り立っているグループもしばしばあった.

- **Group 25 (相談)**
  - 22 歳貯金 400 万ほど女ですね.
  - フランク大学を来月卒業するのですが去年インターンに夢中で就活をしておらず現在就活をしています
  - できるだけ大手に行きたいんですが受かる気はしません
  - 引退や学外活動でガクチカを温かいとエージェントの方に言われているのですが他に大手に受かるための方法はありますか
  - それとも諦めてベンチャーが住所にしまった方がいいでしょうか
  - 今から勉強して慶応の通信に入り直して就活し直すということも考えてます
- **Group 26 (助言)**
  - まああのもう無理であるというのはさっさと諦めた方がいいと思うんですよね.
- **Group 27 (助言)**
  - であのご存知ないかもしれないですけどあの大手企業って今大学 3 年生の就職活動が始まってるんですよ.
  - 要は大学 3 年生の就職活動やってであのもう 4 月とか 6 月とかに大学 3 年生の優秀な人は撮り終わってるんですよ.

4.2 2 つ目の動画 [13] (結果のグループは多いため省略)

動画は, 17 分 40.7 秒に要約された. そして, 一部の助言が直前の相談内容と直接関連していない場合があり, 文脈の流れが断絶することがあった.

4.3 3 つ目の動画 [14] (結果のグループは多いため省略)

動画は, 17 分 53.4 秒に要約された. そして, 助言が連続する場合や, 相談と助言の切り替えが急激な場面では, 時折文脈の整合性が完全に保たれないケースが見受けられた.

5. 考察

実際の実行では, 動画の時間的に約 1/3 以下に要約できたものの, いくつかの重要な課題が浮き彫りになった. 特に, speech-to-text の自然言語処理において, 文脈が通っていない句読点の区切りの文が多く見られた. そのため, 各文にラベルを判定する際に, あまり意味のない文にも相談や助言のラベルが割り当てられてしまい, 結果として不適切な分類が行われている. また, GPT-4o の精度によって要約が過度に行われ, ラベルの文脈に合致していても具体的な内容が削除されてしまっている可能性がある.

これは, アルゴリズムが動画内の複雑な言語的構造や文脈を完全に捉えきれていないことを示している.

さらに, 配信動画の書き起こしの文脈が不整合であることも, 正確な文章の特定を妨げる要因となっていた. これは, 話者の言葉遣いや話し方, さらには背景ノイズや会話の流れなど, 多くの要因によって引き起こされる可能性がある. これらの要因が複合することで, 正確な質問と回答を特定することが困難になっている.

6. 終わりに

本研究では, ファインチューニングした TohokuBERT と GPT-4o を用いた相談・助言形式動画の自動切り抜きシステムを提案した. 提案手法は, 音声抽出から文字起こし,

表 8 グループ別の相談および助言の一覧（時系列順, 上から 6 つのグループを抜粋）

グループ n	ラベル	文 n	開始 (秒)	終了 (秒)	文
1	相談	1	34.5	35.9	なんか久しぶりにやるの
		2	42.5	42.7	大丈夫ですか
		3	69.3	69.9	すいませんね
		4	140.4	142.6	洗濯物干してた
		5	179.8	182.6	大丈夫かな
2	助言	1	210.4	219.6	まあ一応その東京都としてはあの私たちにやってること問題ありましたって自分から言うことはないのでなのである程度あのこういう予想をしてきた人は多いと思うん
3	相談	1	261.9	262.2	はい
4	助言	1	269.8	278.6	オーソドックスなあま IPA のちょっとまああのマイページ自体がもともと癖があるのでそういうことであの癖はあるけどまあ IPA としては普通ってそういう感じですね
5	相談	1	348.6	351.8	それは捏造の可能性あるからちょっと調べていただければどうですか
6	助言	1	351.8	358.5	で終わりでいいじゃないですか, なんか本物だったら辞職するみたいなこと言わなくていいと思うんですよ
		2	372.5	386.8	要はその自分の発言がどういうメリットがあるかデメリットがあるかって考えた時にその売り言葉に乗ることによってファンが増えるとか支持者が増えるとかプラスの要素が僕が見る限り一切ないんですよ
		3	407.5	408.3	僕はありだと思うんですよ
		4	441.7	445.2	まあ炎上マーケティングなことをやった方が得な時期もあるんですよ
		5	449.7	475.8	特に強気なことを言って樹木を集めるのがずらりと思うんすけど今回その強気に行く理由が全くわかんなくてあのあじゃあそのなんか小西さんがさこういうのがありますって言ったらじゃあちょっとわかんないで確認しますって言ってんであの次回確認してから返答しますって言うだけで全然いいと思うんですよでそれで別になんかほらなんか岸田首相が遣唐使って言われるようにその後回しにしますとか確認しますって言ってても何の問題もないじゃないですか

文脈理解, 要約, 動画生成までの一連のプロセスを自動化することで, 動画編集の効率を大幅に向上させることを目指した.

今後の研究では, 提案手法のモデル精度の向上, 処理時間の短縮, 多言語対応, および異なる動画ジャンルへの適用を図ることで, より多様な用途に対応可能な自動動画編集システムの実現を目指すとともに, 定量的な評価を通じてシステムの有効性をさらに明確にする予定である.

参考文献

[1] Andra, A., & Smith, B. (2019). *Automated Lecture Summarization Using Attention-Based RNNs*. **Proceedings of the International Conference on Educational Data Mining**, 45-54.

[2] Bhargavi, R., & Lee, C. (2023). *Text Summarization of Video Content Using Whisper and GPT Models*. **Journal of Artificial Intelligence Research**, 58(2), 123-135.

[3] Gonzalez, M., & Patel, S. (2021). *Lecture Video Summarization with GPT Models*. **Proceedings of the IEEE Conference on Multimedia**, 210-219.

[4] Shambharkar, A., & Kumar, D. (2021). *Extracting Key Sentences from Video Subtitles Using LSA and TextRank*. **International Journal of Computer Vision**, 127(4), 789-803.

[5] Maaz, K., & Zhang, Y. (2022). *Integrating Speech and Visual Frames for Interactive Video Content Generation with GPT-3*. **Proceedings of the ACM Multimedia Conference**, 345-354.

[6] Sainil, R., & Gupta, P. (2023). *Deep Learning Approaches for Audio-Visual Video Summarization*. **IEEE Transactions on Multimedia**, 25(1), 67-80.

[7] Fu, L., & Chen, H. (2021). *Multi-Modal Multi-Level Transformer for Context-Aware Video Editing*. **Proceedings of the AAAI Conference on Artificial Intelligence**, 35(10), 1234-1242.

[8] Huang, T., & Li, X. (2024). *Query-Based Video Summarization Using Textual Cues*. **Journal of Multimedia Computing, Communications and Applications**, 20(3), 301-315.

[9] Sharghi, H., & Thompson, J. (2016). *SH-DPP: A Query-Dependent Video Summarization Model Using Determinantal Point Processes*. **IEEE International Conference on Image Processing**, 1453-1457.

[10] Matsuda, S., & Oi, S. (2024). *CPEX:* 配信動画の視聴者投稿コメントに基づく切り抜き動画の自動制作に関する検討. **情報処理学会 インタラクション 2024 論文集**, 2B-35.

[11] Okazaki, H., Tone, K., Yonamine, A., & Kitahara, T. (2024). *GPT-4* を用いた配信動画中の *Q&A* 切り抜きを目指して. **情報処理学会 インタラクション 2024 論文集**, 3B-33.

[12] 入手先 ⟨https://www.youtube.com/watch?v=NV20jMpDcxA⟩

[13] 入手先 ⟨https://www.youtube.com/watch?v=NGfjgxXgfHk⟩

[14] 入手先 ⟨https://www.youtube.com/watch?v=ZUha6OruO7w⟩