動画とライブ配信のコメントデータを対象とした リアルタイム時系列感性変化計量・可視化・検索システム

尾崎遼太郎† 佐々木史織‡ 清木康†

†慶應義塾大学環境情報学部 〒252-0011 神奈川県藤沢市遠藤 5322

‡慶應義塾大学政策・メディア研究科 〒252-0011 神奈川県藤沢市遠藤 5322

E-mail: †{t15188ro, kiyoki}@sfc.keio.ac.jp, ‡{sashiori}@ sfc.keio.ac.jp

あらまし 動画メディアを対象とした時系列的感性分析は、動画データの新たな共有・検索の主要な機能の一つである。動画共有及びライブ配信を行うウェブサービスの利用者は増加しており、動画及びライブ配信へのコメント投稿は活発である。本システムは、動画共有サービスにおいて動画及びライブ配信を対象として投稿されたコメント（以下コメントデータと呼ぶ）のリアルタイム時系列変化分析により、動画及びライブ配信視聴者の感性変化を計量・可視化・検索する機能を実現する。

キーワード 時系列データ、ストリームデータ、テキストデータ、マルチメディアデータベース、

1.はじめに

近年、動画やライブ配信を共有する Web サービスが発達したことで、インターネット上で大量の動画・ライブ配信メディアが閲覧可能になっている。しかし、これらのメディアの内容に対しての分析はあまり行われておらず、メディア検索はキーワードによる検索に頼られているという現状がある。

動画から受ける感性をもとにした検索や閲覧には大きな需要があり、動画や音楽といった時間的連続性を持つメディアデータを色彩心理学や音楽心理学の知識をベースとして感性的に分析する試みは数多くある[1]-[9]。ただし、動画は大量の画像データと音声データからなるメディアデータであり、これらを制限時間内に、機械的かつ人間の感性に合致するように処理することには限界があると考えられる。

本稿では、メディアデータを多次元空間上にマッピングする手法[1][3]をベースに、現在動画共有サイトにおいて一般的に利用されているコメントデータを活用して動画の感性計量を行う手法を提案する。

また、本手法は軽量なテキストデータを対象とすることで、リアルタイムで放送されているライブ配信に対しても逐次的な処理を行うことが可能であり、ライブ配信に対する分析としても有効な手法であることを示す。

本システムは、動画およびライブ配信の共有サービスにおいてこれらのメディアに対して視聴者から投稿されたコメント（以下、コメントデータと呼ぶ）をもとに、動画の感性を時系列に沿って計量する機能、計量された感性を元に動画同士の距離を計算する機能、計量および距離計算の結果を可視化する機能を提供する。

本システムにおいてコメントデータを活用することには、大きく 2 点のメリットがある。

1 点目として、コメントデータはテキストデータであり、収集や分析がメディアデータに比べて容易であることが挙げられる。動画の特徴を十分表せるだけのデータを集めたとしても、軽量なテキストファイルとして扱うことが可能であり、リアルタイムの分析を行うに当たっても十分に高速に処理を行える。

2 点目として、コメントデータは動画及びライブ配信の共有サイトを利用するユーザーの感性が反映されたデータであることが挙げられる。コメントは、動画やライブ配信の視聴者が、メディアから受けた感性を直接的に反映したものであり、この分析はメディアに対する感性分析手法として有効であると言える。

本稿では、時系列を m 分割、感性 n 種類という設定の下で生成する m 行 n 列の行列を、時系列感性行列と定義した。また、時系列感性行列の計量・可視化・検索を行う方式を時系列感性変化計量方式と定義した。

なお、本システムにおいてライブ配信に対するリアルタイム処理は、一定時間ごとに動画への分析プロセスを実行することによって実現しており、各プロセスにおいては動画とライブ配信によってデータの扱いに違いはないため、以下本稿においては特筆がない限り動画とライブ配信を合わせて動画と呼ぶ。

2.時系列感性変化計量・可視化・検索システム

本システムは、(1)感性語データベース生成機能、 (2)コメントデータを用いた時系列感性行列生成機能、(3)時系列感性行列をもとにした動画間感性的距離計算機能、(4)計算結果可視化機能という、4 機能から構成される。システム構成を図 1 に示す。

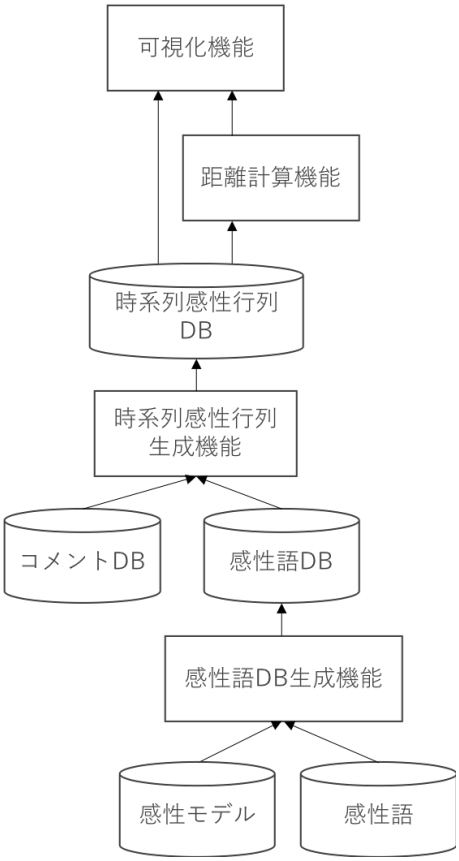

図 1 システム構成図

2.1. 感性語データベース生成機能

本機能は、個人の感性表現に基づいた感性語データベースを構築する機能である。

感性語データベースは、動画の感性を動画のコメントデータにおいて感性を表現する単語（以下感性語）及び、感性ベクトルの集合として定義したものである。

システムには、感性語とその感性語が表現する感性が組み合わせとして入力される。本機能はその感性語をデータベースへ格納するとともに、感性語の感性ベクトルを生成する。

感性語の感性ベクトルは、感性の種類数を n 個とした場合、n 次元のベクトルである。感性モデルに従って二種類の感性ベクトルを生成し、ユーザーの選択によってどちらかを計量に使用する。なお、4.実験方式において述べるように、プロトタイプシステムの実装においては、感性モデルとして、プルチックの感情の輪モデルを使用した[10]。

感性ベクトル生成方式(a)：対応する感性に 1、他の感性に 0 を持ったベクトルである。この方式を使用した場合、最終的な計量結果の取り得る値の範囲は 0~1 の間となる。

感性ベクトル生成方式(b)：対応する感性に 1、その感性と真逆の感性に-1、他の感性に 0 を持ったベクトルである。この方式を使用した場合、最終的な計量結果の取り得る値の範囲は-1~1 の間となる。なお、どちらの生成方式を使用してもシステムにおけるその後のプロセスに違いはない。

ライブ配信のリアルタイム分析に当たっては、本プロセスを、リアルタイム分析を行う前に一度だけ実行し、感性語データベースを作成する。

2.2.時系列感性行列生成機能

本機能は、コメントデータベースと感性語データベースを対象として、時系列感性行列を生成する機能である。

図 2 に示すように、本機能により生成される時系列感性行列は m 行 n 列の行列である。このとき、m は時系列を分割したブロック数であり、n は感性の種類数である。

感性
$f_1, f_2, \ldots f_n$

時系列分割
t_1
t_2
$\vdots$
t_m

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}$$

図 2　時系列感性行列

本システムは対象の動画を等時間で分割し、動画を各ブロックの感性ベクトルの配列として捉えることで時系列を扱う。この際の分割粒度は、ユーザーが動画の感性変化を反映するのに適すると考える値が、システムに入力されたものを使用する。時系列感性行列を生成する際の時系列分割のブロック数 m は、入力された時系列分割粒度によって対象の動画を分割した個数である。すなわち、時系列感性行列の行数は対象の動画の長さによって動画ごとに異なる

時系列感性行列は、コメントごとに感性ベクトルを生成し、コメントの感性ベクトルを時系列に沿って計量することによって生成する。

コメントの感性ベクトルは、コメントに含まれる感性語の感性ベクトルを合計したのち、2 ノルム正規化を行ったものである。すなわち、コメントの感性ベクトル($\vec{c}$)は、感性の種類数を n とした場合、n 次元かつ長さ 1 のベクトルであり、コメントに含まれる感性語の感性ベクトル($\vec{w}$)の個数を p とした場合、図 3 の式によって表される。

$$\vec{c'} = \sum_{k=0}^{p} \vec{w_k}$$

$$\vec{c} = \frac{\vec{c'}}{|\vec{c'}|}$$

図 3　コメントの感性ベクトル計算方法

このようにして生成したコメントの感性ベクトルを、そのコメントが属する時系列ブロックごとに合計し、ブロックの感性ベクトルを生成する。すなわち、あるブロックに含まれ

るコメントの数を q とした場合、ブロックの感性ベクトル($\vec{b}$)は図 4 の式で表される。

$$\vec{b} = \sum_{k=0}^{q} \vec{c_k}$$

図 4　ブロックの感性ベクトル計算方法

このようにして生成したブロックの感性ベクトルを時系列順に並べた行列を生成し、その行列をコメント数で割り、行数を乗じた行列がその動画の時系列感性行列である。すなわち、ある動画の時系列感性行列(A)は、時系列分割数 m、動画のコメント数を r とした場合、図 5 の式によって表される。

$$A = \begin{pmatrix} \vec{b_1} \\ \vec{b_2} \\ \vdots \\ \vec{b_m} \end{pmatrix} \times \frac{m}{r}$$

図 5　時系列感性行列計算方法

最後に、このようにして生成した時系列感性行列を他の動画情報メタデータと紐付けて格納した時系列感性行列データベースを作成する。

ライブ配信のリアルタイム分析に当たっては、一定時間おきにこのプロセスを実行し、時系列感性行列を更新する。

2.3.時系列感性行列をもとにした動画間感性的距離計算機能

本機能は、時系列感性行列同士の感性的な距離を計算する機能である。

本システムにおいて、距離計算の方式（以下メトリックと呼ぶ）は複数定義されており、そのうちユーザーから選択された一つを実際に計算し、動画同士の感性的な距離として扱う。

本システムにおいてメトリックは全て、同じ大きさの行列二つを入力とし、それら二つの行列同士の距離を何らかのアルゴリズムによって計算し、取り得る値の範囲を 0~1 に正規化して出力する関数として定義されている。これにより、ユーザーがどのメトリックを選択したとしても、前後のプロセスを同じように実行することが可能になる。

一方で、本システムにおいて時系列感性行列は、動画の長さによって大きさが異なっており単純に比較することはできない。そのため、本システムで距離計算を行う際には、対象の動画の一部分を抽出し、他の動画も同じ長さの部分をずらしながら抽出していき、それぞれ距離計算を行ったうちの最小値をこれらの動画間の距離と定義する。すなわち、時系列感性行列 A の p ブロック目から q 個の部分を対象とした距離計算において、m 行 n 列の時系列感性行列 B との距離 (d)は、選択したメトリックにおける距離計算を f(X,Y)とした場合、図 6 の式によって表される。

$$d = \min\{f(A[p:p+q], B[x:x+q]), x = 0, 1, \cdots, m-q\}$$

図 6　時系列感性行列同士の距離

ライブ配信のリアルタイム処理に当たっては、時系列感性行列の更新を行った際に本プロセスを実行する。この時、ライブ配信のうち計算に用いる長さをユーザーが設定しておき、現時刻からその長さまでの時系列感性行列を、その時点でのライブ配信を表現する時系列感性行列として扱う。

2.4.計算結果可視化機能

本プロセスは、時系列感性行列と、それをもとに計算された動画同士の感性的な距離を可視化するものである。

距離の可視化は、対象の動画からの距離によってそれ以外の全ての動画をソートした結果を出力することによって行う。

ライブ配信のリアルタイム処理に当たっては、一定時間おきに本プロセスを実行し、表示を更新する。

3.実現方式

プロトタイプシステムを以下の環境と方式により Python を用いて実現した。

対象とする動画共有サイトとして、今回はニコニコ動画を利用した[11]。理由としては、利用者が多くデータが豊富であること、時系列に沿った形でコメントされるシステムであること、データが配布されており利用がしやすいことがあげられる。

対象とする動画のデータとして、国立情報学研究所にて提供されているニコニコデータセットを利用した[12]。このデータは、動画情報およびコメントデータが JSON 形式で表現されたデータである。

本システムではコメントデータの少ない動画を対象とした場合、あるいは著しく長いもしくは短い動画を対象とした場合に、計量が適切に行われないことが想定される為、それぞれ採用する動画の長さやコメントデータ量に閾値を設定し、コメントデータが閾値を超え、動画の長さが範囲内に存在する動画のみをリストアップするプログラムを作成した。

ライブ配信のリアルタイム分析に際しては、大きく２つの方法でコメントデータを取得した。

一つ目はライブ配信サービスであるニコニコ生放送[13]を使用した方法である。この方法では、ニコニコ生放送のコメントサーバーにアクセスし、websocket 通信を用いてリアルタイムでコメントデータを取得するプログラムを作成した。

二つ目は SNS の一つである Twitter[14]のハッシュタグ検索結果を使用した方法である。Twitter 自体はオープンな SNS であるが、近年ライブ配信に対して専用のハッシュタグを用いて感想を投稿することは一般的であり、Twitter のハッシュタグ検索結果はライブ配信へのコメントデータとして十分利用可能であると考えた。この方法では、Twitter の API を利用

し、一定時間おきにTwitterのハッシュタグ検索結果を取得するプログラムを作成した。実装はTwitterAPIにアクセスを行えるPythonのライブラリであるtweepy[15]を用いて行った。

本システムの入出力は、Pythonのwebフレームワークである Django を用いて、ブラウザから行えるように実装した[16]。

3.1. 感性語データベース形成機能の実現

ユーザーはブラウザを使用し、感性と対応する感性語を入力することが可能である。本プロセスは入力された値を元に、感性語データベースを CSV 形式のファイルとして保存する。

プログラム内において感性語の数、感性の種類数など、感性を表現する値は全て変数として管理されており、可変的である。

3.2.時系列感性行列生成機能の実現

時系列分割に使用するブロックあたりの長さは、ブラウザから入力することが可能であり、プログラム内では変数として管理する。そのため、分析の際の時系列粒度は自由に変更が可能である。

行列計算は Python のライブラリである NumPy を用いて実装した[17]。時系列感性行列は、それぞれの動画において NumPy のデータ保存形式である npy ファイルとして保存する。

本機能において、今回実装の対象としたニコニコ動画特有の問題として、強い感情を持つわけではないが広く使用される文言の存在がある。例えば、動画の最初に挨拶の意味合いで投稿される「うぽつ」や、シリーズ系の動画に投稿される「追いついた」などは、強い意味を持たないものの頻出し、時系列感性行列に強い影響を与え、異常値となってしまう恐れのある単語である。今回の実装では、こういった単語をリスト化し、その単語が出現した場合にはそのコメントのスコアを0とするプロセスを実装した。

3.3.時系列感性行列をもとにした動画間感性的距離計算機能の実現

プロトタイプシステムにおいて、本機能にはメトリックが二種類実装されている。

一つ目は、行列同士のユークリッド距離を求め、最大ブロック数の２倍で割って正規化を行った値を返す関数である。すなわち、m 行 n 列の行列 A と B は m*n 次元のベクトル$\vec{a}$、$\vec{b}$として扱い、その距離 d_{Euc} は、選択した動画郡において起こり得る最大ブロック数が p 個の時、図 7 の式によって表される。

$$d_{-Euc} = \frac{\sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 \ldots (a_{mn} - b_{mn})^2}}{2p}$$

図 7　ユークリッド距離を使用したメトリック

二つ目は、行列の列同士のコサイン類似度を求め、それらの平均値を 1 から引いた値を返す関数である。すなわち、m 行 n 列の行列 A と B の距離 d_{Cos} は、図 8 の式によって表される。

$$d_{-Cos} = 1 - \frac{\sum_{i=0}^{n} \cos(\vec{a_i}, \vec{b_i})}{n}$$

図 8　コサイン類似度を使用したメトリック

本プロセスにおいて使用するメトリックの選択も、ブラウザから行うことができる。

本プロセスは、検索元の動画からそれぞれの動画までの距離を、選択されたメトリックによって計算し、動画番号と距離の組み合わせを csv 形式で保存する。

csv ファイルを扱う際には、Python のライブラリである pandas を用いた[18]。

3.4.計算結果可視化機能の実現

各動画の時系列感性行列は npy データから読み込みグラフとして可視化する。グラフの可視化にはjQueryのグラフ描画ライブラリである chart.js を用いた[19]。

動画の距離は csv データから読み込み、距離の近いいくつかの動画と合わせて画面に出力を行う。

4.実験

ニコニコ動画に存在する動画全てを対象とした場合、動画数が多くなりすぎること、使用される言葉や頻出する固有名詞が時代によって異なることなどから、計量が正しく行われないことが想定されるため、データセットの中で比較的新しい動画群を対象として計量を行うこととした。

まず、自分の感性モデルとして、プルチックの感情の輪モデルを使用した[10]。プルチックの感情の輪モデルでは、８つの感性(”joy”, ”trust”, ”fear”, “surprise”, “sadness”, ”disgust”, “anger”, “anticipation”)が定義されており(図 9)、今回はこの８つの感性を基準として動画を分析することとした。

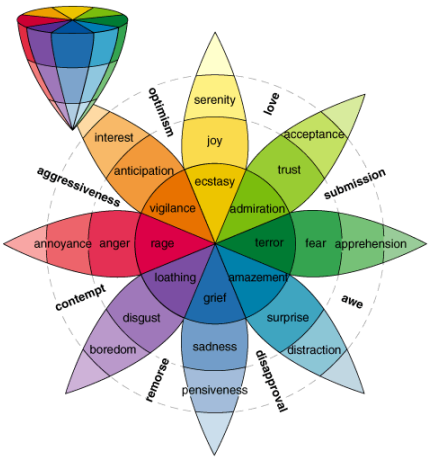

図 9 プルチックの感情の輪モデル

形態素解析エンジンである MeCab[21]及び、固有名詞やネット特有の表現を分類することを可能にする MeCab 辞書である mecab-ipadic-neologd[22]を利用し、ニコニコ動画のコメントにおいて頻出する単語の中で、感性を表すと考えられるものを計 70 個リストアップし、先ほど定義した 8 つの感性を感性語の集合として表現した（図 10）。

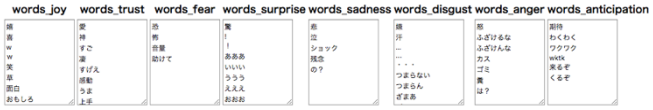

図 10　感性語データベース作成のウェブインターフェース

それを元に、それぞれの感性語を 8 つの感性のベクトルとして表現した、感性語データベースを構築した(図 11)。

	joy	trust	fear	surprise	sadness	disgust	anger	anticipation
嬉	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
喜	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
w	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
w	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
笑	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
草	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
面白	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
おもしろ	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
愛	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0

図 11 感性語データベース

分析の対象となる動画は、コメントの閾値として 500 個、動画の長さの閾値として 5 分以上 30 分未満という設定を行い、リストアップを行った。今回の実験では 62 個の動画がこれに該当した。対象となった動画の一部を図 11 に示す。

0 sm29560079
【ボイスロイド実況】茜のカービィボウルをプレイするで！part10
1 sm29560199
【MUGEN】春閣下 12P 前後 狂下位ランセレ！台パンの向こう側へ！Part40
2 sm29560445
【ゆっくり実況】UNDERTALE Part19 (Genocide End)
3 sm29560591
【実況】 いまだかつてないほど初見すぎる Minecraft Part70
4 sm29560791
週刊 VOCALOID と UTAU ランキング　#465・407
5 sm29560898
「サタン」と「ガブリエル」の配布に物申す。【シャドウバース】
6 sm29561313
【ゆっくり TRPG】超人高校生＋おまけのやりたい放題クトゥルフ　59話
7 so29560248
【ペカる TV】群馬の優良店で 10 万円取り戻したいんだの巻（牙狼金色・魔戒・沖ドキ等）【それ行け養分騎士 vol.17】

図 12　対象となった動画の一部

時系列分割時間は 30 秒と設定した。
以上を設定した上で実際に計量を行った。
各動画の時系列感性行列の例を、図 13、図 14 に示す。

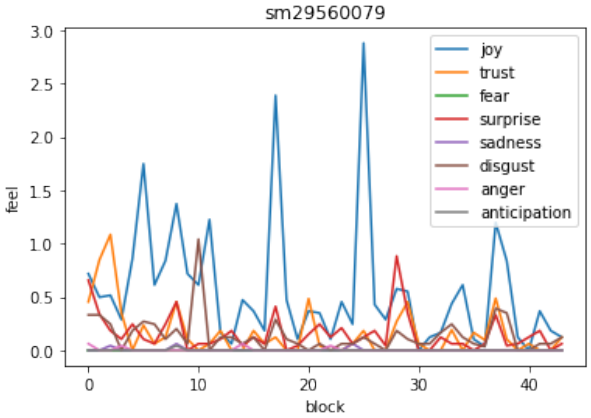

図 13　動画(id : sm29560079)の時系列感性行列

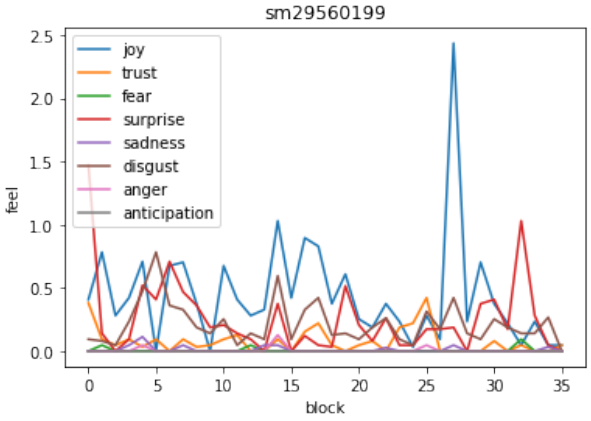

図 14　動画（id : sm29560199）の時系列感性行列

また、検索元動画として動画 ID:sm29560079 の動画、抽出する部分として 5 ブロック目から 5 個の部分、メトリックとしてコサイン類似度を設定し、動画同士の距離計算を行った結果を図 15 に示す。

video_id	block_id	distance	
746	sm29554027	3	0.011857
1359	sm29558592	37	0.020488
1428	sm29559971	16	0.020675
1528	so29553397	44	0.020946
41	sm29550166	41	0.022889
1346	sm29558592	24	0.023441
767	sm29554027	24	0.023840

図 15　動画同士の距離計算結果

図 15 は、上記の検索元動画に対して最も感性的に近い動画は動画 ID:sm29554027 の 3 ブロック目から 5 個の部分であることを示している。
以上をもとに、実際にブラウザ上に出力した結果を図 16 に示す。

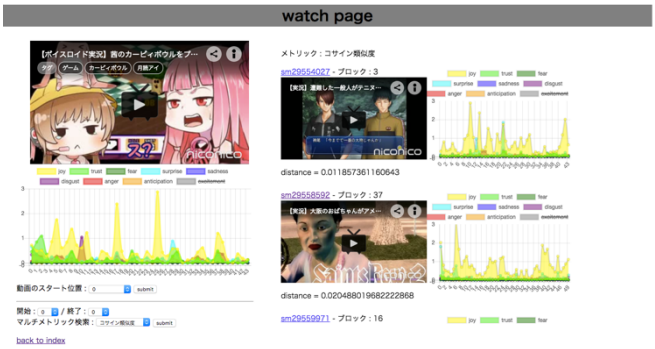

図 16　ブラウザ上での出力結果例

続いてライブ配信の分析を行った。今回はニコニコ生放送にて放送されていたライブ配信 ID:lv317897874 のライブ配信を対象に計量を行った。

本ライブ配信は、人気ゲームの公式配信であり、視聴者数およびコメント数が計量を行う上で十分であると判断した。

本ライブ配信をリアルタイムに分析し、結果をブラウザ上に出力したものを図 17 に示す。

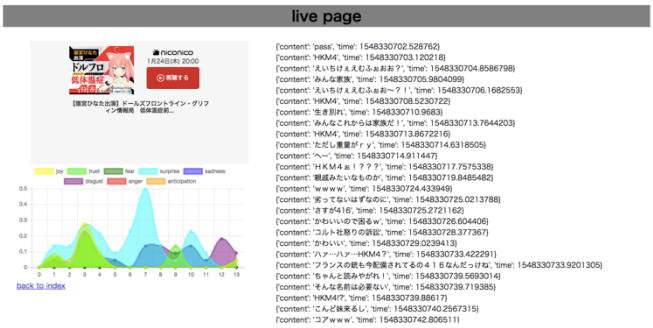

図 17 ライブ配信分析の出力結果例

5.考察

今回の実験において、時系列感性行列は概ね動画の感性を表現していると言えるものになっていた。また、検索結果も人間の感性とマッチしていると言えるもので、不自然な検索結果が表示されるような場面は少なかった。

リアルタイム分析に関しても、計算結果自体は完全ではないものの、人間の感性と近いになっていると感じた。

一方で課題としては、web システムとしての完成度の低さがあげられる。本システムは個人差を計量する方式を実装するものであるが、web システムとして、個人の情報を別々に保存する機構が実装できていないため、個人の感性を反映するという点で、まだ不十分であると考えられる。

また、アルゴリズムの最適化や処理の高速化が不十分である点もあげられる。現在の実装は 62 件の動画という狭い範囲に絞って行っているため、十分高速に処理が完了しているものの、データとして活用できる動画の数はそれよりもはるかに多いため、現在の処理時間は不十分であると考えられる。特に、リアルタイム分析では逐次的に処理が行われるため、制限時間が厳しく制限されており、処理速度を速くしなければデータ量を増やすことができないため、システムとしての拡張性が大きく損なわれてしまう。

感性的に遠い動画を検索する必要性が薄いことや、いくつかの計算が行列計算によって数学的に変形できることなどを踏まえると、高速化が可能な点はいくつかあると推測できるため、今後実装を進めたい。

6.結論および今後の展望

本稿では、動画を感性的に計量する手法として、動画共有サイトのコメントデータを活用することを提案し、時系列感性行列を利用した実際の計算手法と実現方式について述べ、ライブ配信のリアルタイム分析を行い、結果を示した。

今後の展望としては、時系列変化のデータを蓄積させていき、機械学習のような学習やデータマイニングを活用し、モデル化を行うことが想定される。本システムによって動画を数値の行列として表現することが可能となっているため、パターン抽出や予測モデルの作成などを行うことも可能ではないかと考えている。

参　考　文　献

1. 清木康，金子昌史，　北川高嗣，″意味の数学モデルによる画像データベース探索方式とその学習機構，″ 電子情報通信学会論文誌，Vol. J79-D-II，　No. 4，pp. 509-519，Apr. 1996.
2. S. Dubnov, Y. Kiyoki, “Opera of Meaning: film and music performance with semantic associative search”, Frontiers in Artificial Intelligence and Applications, Information Modelling and Knowledge Bases XX, Volume 190, pp. 384 ? 391, 2009.
3. Kurabayashi, S., Ueno, T., Kiyoki, Y.: A Context-Based Whole Video Retrieval System with Dynamic Video Stream Analysis Mechanisms. In: 11th IEEE International Symposium on Multimedia, San Diego, USA, pp. 505-510 (2009).
4. Kurabayashi S., Kiyoki Y. (2010) MediaMatrix: A Video Stream Retrieval System with Mechanisms for Mining Contexts of Query Examples. In: Kitagawa H., Ishikawa Y., Li Q., Watanabe C. (eds) Database Systems for Advanced Applications. DASFAA 2010. Lecture Notes in Computer Science, vol 5982. Springer, Berlin, Heidelberg.
5. Ijichi, A. and Kiyoki, Y.: "A Kansei Metadata Generation Method for Music Data Dealing with Dramatic Interpretation," Information Modelling and Knowledge Bases, Vol.XVI, IOS Press, pp. 170--182, May, 2005.
6. Jeremy K. Hall, Shuichi Kurabayashi, Yasushi Kiyoki, Multimedia Data Analysis on a Massively Distributed Parallelization Network of Anonymous Web Clients, Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference, 20-23 April 2010.
7. Munehiro Fukuda, Shuichi Kurabayashi, Jeremy K. Hall, and Yasushi Kiyoki, "Morphing Parallelization Strategy to Support On-the-Fly Video Analysis," In Proceedings of the 2010 International Conference on Parallel and Distributed

Processing Techniques and Applications - PDPTA 2010, pp.541-547, Las Vegas, USA, July 12-15, 2010.

8. 高橋 勝稔, *濱田 賢吾, 馬野 元秀 「言葉による表現を用いた類似時系列検索の診断への応用」 第 31 回ファジィシステムシンポジウム セッション ID: TD4-2 2016/02/26.
9. 廣安 知之, 福島 亜梨花, 山本 詩子, 横内 久猛 「2 本の時系列データの類似部分自動抽出法の提案 ―fNIRS 時系列データに対する検討」情報処理学会論文誌数理モデル化と応用（TOM） 7 巻 2 号 ページ 64-73 発行年 2014-11-27.
10. R. Plutchik, "The Nature of Emotions", Am. Scientist, vol. 89, pp. 344, 2001.
11. ニコニコ動画 http://www.nicovideo.jp
12. 国立情報学研究所 ニコニコデータセット https://www.nii.ac.jp/dsc/idr/nico/nico.html
13. ニコニコ生放送
14. Twitter https://twitter.com/
15. Tweepy http://www.tweepy.org/
16. Django https://www.djangoproject.com/
17. Numpy http://www.numpy.org/
18. Pandas https://pandas.pydata.org/
19. Chart.js https://www.chartjs.org/
20. MeCab http://taku910.github.io/mecab/
21. mecab-ipadic-NEologd https://github.com/neologd/mecab-ipadic-neologd