2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC) | 978-1-6654-2463-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/COMPSAC51774.2021.00065

PeakVis: a Visual Analysis Tool for Social Network Data and Video Broadcasts

Pedro Henrique M. Sanvido[1], Gabriela B. Kurtz[2], Carlos R. G. Teixeira[2], Pedro P. Wagner[1],
Lorenzo P. Leuck[2], Milene S. Silveira[1], Roberto Tietzmann[2], Isabel H. Manssour[1]
Pontifical Catholic University of Rio Grande do Sul, PUCRS.
School of Technology[1] - School of Communications, Arts and Design[2]
Av. Ipiranga, 6681 - 90619-900, Porto Alegre, Brazil
Email: {pedro.sanvido,carlos.teixeira85,pedro.wagner00,lorenzo.leuck}@edu.pucrs.br
{gabriela.kurtz,milene.silveira,rtietz,isabel.manssour}@pucrs.br

***Abstract*—Television and live internet broadcasts coexist with a stream of social network commentary, engaging audiences in more complex ways than traditional ratings can represent, demanding convergent tools to analyze such contemporary media. In this work, we present PeakVis, an interactive tool that syncs a video recording with a responsive line graph representing the total number of tweets from each moment, the top messages, a dynamic word cloud, and a semantic graph showing word correlation. Thus, this novel interactive approach allows the analysis of broadcast highlights identified through Twitter posts peaks. We explored this freely available tool's applicability by analyzing two different case studies: the season finale of a reality show and the final episode of a telenovela. We could observe that it allows us to quickly identify the most relevant segments and grasp the discussed subjects in-depth through the obtained results, showing its value in broadcast analysis.**

I. Introduction

Nowadays, television broadcasts are expected to be accompanied by a stream of social media postings by the audiences, mediating the asymmetrical relationship between the television network and spectators through commentary, humor, and criticism. Social networks, such as Twitter, have been used as a way of interaction with the audience. It has allowed them an opportunity to contribute to the media sphere [1] and for people to express themselves about a program or event [2].

To help us to understand this scenario, we designed and developed PeakVis, an interactive tool able to select and analyze broadcast highlights based on the traffic peaks on Twitter. It allows the user to sync and scrub a video recording to a line chart plotted from a social media dataset, viewing dynamically the most retweeted messages, a word cloud, and a semantic graph presenting the most connected words. The word cloud and the graph allows a semantic analysis of the content, showing the frequency distribution and the connection of words over time. Consequently, they help in context understanding and reveal the main topics covered. The tool's implementation results from a joint effort of researchers from the Computer Science and Communication fields.

As a prototype, the PeakVis tool was validated through two case studies. The first was the finale of the 2018 season of the Big Brother Brazil reality show, where the audience voted on which of the four confined contestants would win the prize. The finale had the highest audience since the 2011 edition, after years of constant decline. The second was the final episode of "A Dona do Pedaço" (or "The Boss of the Block"), a telenovela that aired in 2019, representing the top primetime fictional product of Brazilian television.

The main contribution of PeakVis is an innovative approach for audience behavior analysis, including:

- An interactive way to analyze broadcast highlights based on the peaks of traffic on Twitter. PeakVis allows the user to explore a video recorded and its related tweets through the synchronous interaction with the video itself, choosing the peaks he/she wants to deepen analyze and also increasing or decreasing the sensitivity in which these peaks are detected. It allows both a panoramic and focused analysis of semantic aspects through a dynamic word cloud and a semantic graph of the correlation between the words.
- A generic approach that can combine broadcasts programs or live-streamed events in general and collect tweets from different knowledge fields. PeakVis can be applied for different purposes and professionals, e.g., supporting media analysts in understanding the impact of certain events on social media sites, enabling producers to content summaries, aiding advertisers in choosing the best moments to advertise for the general public.
- The validation of PeakVis through the analysis of two case studies, highlighting not only possibilities for its improvement and novel uses but also the study of these specific cases *per se* and the impact of specific events on their audience.
- Its availability to be used for free by the academic and professional community. PeakVis, in its current version, is available – for free at GitHub[1] - for use for whoever is interested in a broadcast analysis.

The remainder of the work is organized as follows. Section II presents related work. Section III describes PeakVis and how the highlights are identified. Section IV presents a discussion

[1]Available at https://github.com/DAVINTLAB/Peakvis. The tool and its documentation are in continuous development. The feedback of its users is essential to help the research team to improve it.

978-1-6654-2463-9/21/$31.00 ©2021 IEEE
DOI 10.1109/COMPSAC51774.2021.00065

about the two real-case studies. Finally, Section V presents our conclusions, as well as future work possibilities.

II. Related Work

We looked for research that had highlights extraction, social media, and audiovisual analysis as themes. We also search for works that analyzed audiovisual productions with other variables, to understand different approaches. Some of the analyzed works were written and developed before Twitter (launched in 2006) or other social media websites were created or became widespread. For example, Harb and Chen [3] proposed to test automatic sports highlights detection based on an audio classifier. Hanjalic [4] developed a video abstraction algorithm to extract sports videos highlights based on what the fans reacted most to. These studies are interesting to observe that the video reactions and audio matter, and they can be combined with other methods, such as the use of Twitter data. The next subsections present the works that use Twitter data.

A. Analyzing sports videos through highlights detection

Regarding the use of tweet datasets, several of the analyzed works were only related to sports video highlights. The work developed by Langan and Smeaton [5] used a filtered stream of tweets to distinguish highlights in sports transmissions automatically. Their filtered search and content analysis shows efficiency in detecting significant events and generating a summary for associated media, proving that it is possible to use tweets to find points of interest and audience talking points. A tool called #EpicPlay was presented by Tang and Boring [6] to also select video highlights of live sports events (e.g. American football) through the reactions of sports fans on Twitter. The researchers separate the incoming tweet stream by home/away teams and demonstrate that this approach can be used to select highlights specific for fans of each team.

Also in the sports scenario, Jai-Andaloussi et al. [7] propose a summarization approach that combines video content analysis (score box analysis and audio energy activity) and social media streams (detection of events in the live Twitter feed and highlight detection by using Twitter event). They presented tests showing the efficiency of the proposed approach based on a case study and concluded that it can improve the quality of soccer video summarization and could be used in other sports too. In the case of Doman et al. [8], the presented technique is grounded on video recording and Twitter data collection. The authors conceived a metric they named Twitter Enthusiasm Degrees and demonstrated that their method could work for English, Japanese, Spanish, and Italian tweets. Likewise, Fião et al. [9] create video highlights based on the emotions of the fans, but they also considered audio, movement data extracted from the video, and emotion data. In a continuation of this work, they presented a positive preliminary evaluation [10].

Koleejan, Takamura and Okumura [11] also propose a summarization approach using social media streams, including as one of its steps a burst detection approach. They presented tests showing the efficiency of the proposed approach based on ten matches from the English Premier League and Champions League in 2017.

B. Analyzing sports videos through events detection

We also found works that aimed not to produce sports highlights, but to detect certain events during the sports matches, such as goals (football) or aces (tennis). Zhao et al. [12], e.g., investigated real-time sensing for frequent events on Twitter, such as football games. They present a solution based on content analysis, but it was not a fully automated solution because it demanded human selection of significant context words. Hsie et al. [13] worked detecting highlights and extracting semantic from events in multiple sports by making use of user-generated tweets. They trained models for each sporting domain and used text analysis methods to classify events for multiple sports. Their work detects goals in soccer games with remarkable high accuracy, but events such as “aces” in tennis are harder to predict. Similarly, Oorschot et al. [14] approached this theme specializing in soccer games. Their method differs in which they go beyond just event detection, given that with the resulting set of demarcated events, they employed natural language processing and machine learning methods on the tweets to classify what happened in those events, as the occurrence of goals and red cards.

Huan et al. [15] present a participant-centered social event summarization using Twitter streams. They focus first on participant detection, and then on sub-event detection and summary tweet extraction. The authors highlight this approach allows us to easily identify important events related to each participant.

C. Analyzing sports videos in general

Other works can even be applied with different contexts but have been also tested with sports datasets. Hannon et al. [16] presented PASSEV, a prototype system that uses real-time web data as the basis for event detection and summarization of video streams. It slices a video into a set of segments and indexes them by the content of the related tweets. The method presented by Nakazawa et al. [17] proposes to detect significant scenes in TV programs (baseball games) and automatically annotate their content through Twitter analysis. The annotation was focused on personal names frequently appearing in these tweets and the keywords co-occurring with them. Hayama [18] used tweets classified according to users' behavior to improve the detection of TV Program highlight scenes. For him, Twitter users behave in different ways (e.g., conversing or sharing information), and this behavior can affect the correct detection. The author focused on soccer game TV programs.

D. Final analysis

It is possible to notice that most of the presented works are focused on sports events. Some of them consider events in their processing, such as goals and red cards [14], or text features such as repeated characters for GOOOOOOAAAAALLL [8], which are specific to sports videos, making their use in other contexts unfeasible. As for #EpicPlay limitations, some typical user behaviors make it difficult for the highlight tool to reflect an exciting moment of the game. Moreover, it does not go further to understand those users, it was designed just for live broadcast sports games, and it does not provide a synchronous presentation of the tweets with the video. Other works include video or audio analysis, e.g. [9], which may require more processing and difficult its use for any type of video broadcast. Lastly, the tools used as well as the videos extracted were not made available for other researchers to replicate those models or to analyze and compare their results (e.g. [5] and [13]).

We did not find any recent research designed for different events and with the possibility to synchronize peaks of tweets and word clouds with the exact moment of the video. We understand that the PeakVis tool is unique as an enabler of an in-depth analysis of audience behavior, allowing the user to specify the sensitivity to detect highlights, and automating part of the syncing between data and video while leaving room for human-centered contextual commentary. Furthermore, PeakVis provides a graph with word correlation and can be used with different video broadcasts, like sports, reality shows, and the launching episodes of popular series, for instance.

III. The PeakVis Tool

PeakVis is built upon the primary idea of analyzing online traffic generated in response to televised broadcasts. For this, PeakVis uses two data sources: (1) a video recording file of a given broadcast; (2) a dataset of timestamped Tweets matching the runtime of the provided video. With both data sources, the PeakVis tool synchronizes the provided video recording with a line chart of Tweets per second, marking the current playtime on the time grid and displaying the contents of the most relevant Tweets at the given time. PeakVis also implements interaction features commonly seen in video editing software, such as zooming in and out of the timescale, and scrubbing the video recording back and forth, which happens coordinately and responsively in all views. Furthermore, PeakVis provides a dynamically updated word cloud, which displays a summary view of the Tweets' contents until the analyzed point in time and a semantic graph with the most connected words.

Lastly, inspired by related work [12], [6], [13], [8], PeakVis implements a simple and customized model of highlight detection. Along with the line chart view, the tool highlights traffic peaks, which can be interpreted as the most appealing or most commented moments for the crowd. The user can customize the threshold in which these highlights are detected, increasing or decreasing the sensitivity for which they appear.

By analyzing the contents of the views and interacting with the features, users can intuitively understand which moments generated the most impact on social media sites and understand which events generate more online buzz.

A. *PeakVis Interface*

Figure 1 displays the main interface of the developed prototype. Through the input forms at the top (Figure 1a), the user can upload a compressed MPEG-4 video and a corresponding time-stamped JSON file that matches the video file. Two preprocessed text files are also necessary and will be loaded simultaneously as long as they match the filename of the original JSON dataset (see next section for details).

The line chart plotting the volume of Tweets at each second in time is under the input forms (Figure 1c). To facilitate video navigation, already watched segments are colored blue instead of the default unwatched grey lines. Moments that precede highlights are marked with the purple circle, which indicates a traffic peak is coming up. The "highlight" button above the line chart allows the user to navigate through the identified highlights more quickly. Both the video and the line chart will jump to the next identified highlight by pressing the button. The highlight slider located at the top left (Figure 1b) indicates the current setting for the highlight detection sensitivity. Higher sensitivity settings will only mark the most expressive peaks in online traffic volume, such as the ones indicated by Figure 1. Lower sensitivity settings will translate into behavior that marks more segments as highlights, as long as these segments are above the lower highlight detection threshold specified.

Figure 1d shows the playback view, which is a video player for the uploaded video. It is important to mention that the user can also use a YouTube video instead of uploading their own MP4 file. This is done by pasting the video URL in the video form instead of typing down an MP4 file name. Doing so will load the YouTube embed player as opposed to the HTML 5 player. However, both players function in the same way for PeakVis. Any actions made in the player, such as playing, scrubbing, and pausing the video, will be reflected in the line chart, the comments view, and the word cloud. The comments view (Figure 1e) outputs the textual contents of the most retweeted Tweets that occurred at every second.

Finally, below the playback view (Figure 1f) is displayed the word cloud and on its side the semantic graph (Figure 1g). The word cloud is dynamically updated, i.e., it displays the most commonly used terms in the user-provided Tweets dataset, from the beginning until the video moment that is being watched. The words are updated and organized according to the frequency in which they appear in the tweets. At the beginning of the video, the word cloud will be empty, but as it updates itself cumulatively in every time frame of 15 seconds, by the end of the video, the word cloud should represent the entire corpus of data. The word cloud also updates accordingly to video scrubbing events (which includes highlight jumping), therefore corresponding to the exact timeframe selected by the user at any given time. In this way, it should enable analysts to quickly detect which words generate the biggest online reaction and identify which terms grew in usage according to the observed events. To expand the semantic exploration of the word clouds, not only

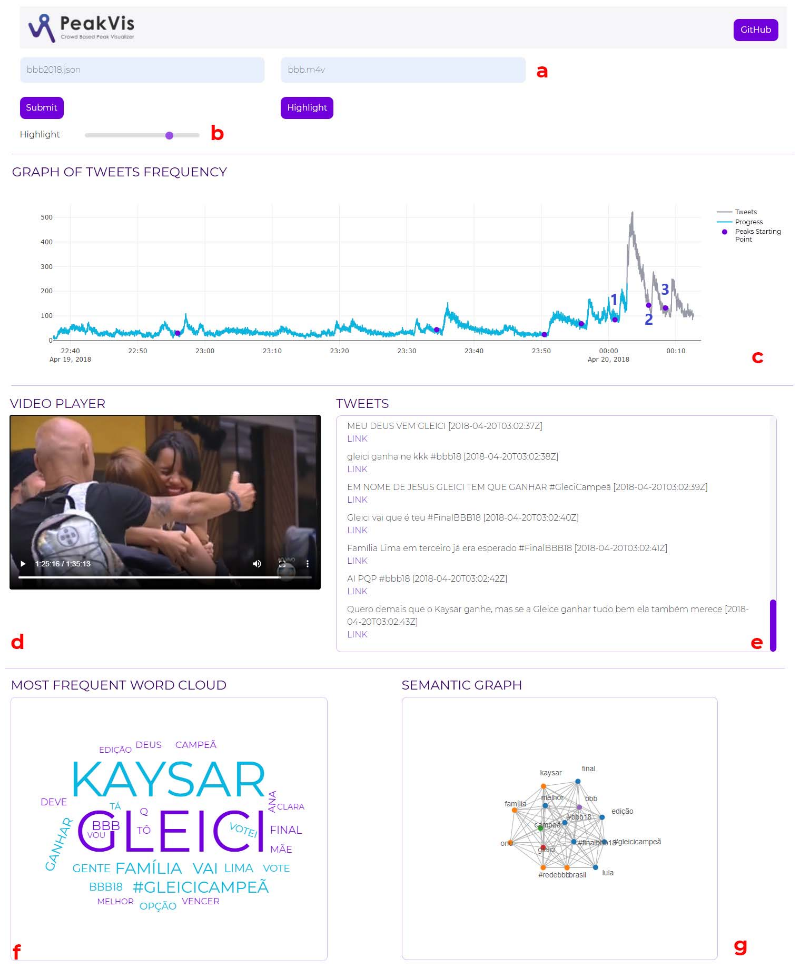

Fig. 1. PeakVis interface: (a) input file names for uploading; (b) slider for highlight sensitivity configuration; (c) line chart plotting the total number of Tweets at each second (blue represents the part of the video already seen); (d) playback video view; (e) most retweeted Tweets; (f) word cloud of the most commonly used terms; (g) semantic graph showing words correlations.

presenting the most used words but the context in which they are inserted from their connections, PeakVis also provides a semantic graph showing their correlations. To capture the most representative posts, the semantic graph factors only the 1000 most retweeted tweets. While the semantic graph does not react to interactions in the line charts or video events, it does allow for a mouse-over interaction, in which users can highlight the "subnetworks" contained in the graph.

B. PeakVis development

PeakVis is developed using Web technologies and is characterized as a single-page Web application. It runs locally on a host computer and is not remotely hosted elsewhere. Once a user loads up the main interface, no other pages are loaded, and the main interface is only responsively updated according to the triggered events.

Figure 2 illustrates a high-level view of how the implementation is structured at a software level. The user provides the input files through a Python Flask[2] HTTP server, which is hosted locally on the host computer. The minimalistic web framework was chosen to create more direct interfacing with the Python programming language, which is invoked through the Web interface to perform some preprocessing-related tasks, which are detailed later in this section. The user-provided data is computed using Javascript code ran in Node.js[3] runtime environment. This technology was chosen for its efficiency and scalability compared to regular Javascript, which is desirable when dealing with large amounts of data, such as Tweet datasets. Moreover, its blocking features allow us to implement more easily synchronized interactions along with the playback view and the line chart. The playback view is handled using the default browser player, and its corresponding HTML 5[4] functions for managing interactions. Regarding layout and

[2]Flask: https://flask.palletsprojects.com/en/2.0.x/
[3]Node.js: https://nodejs.org/en/
[4]HTML standard: https://html.spec.whatwg.org/multipage/

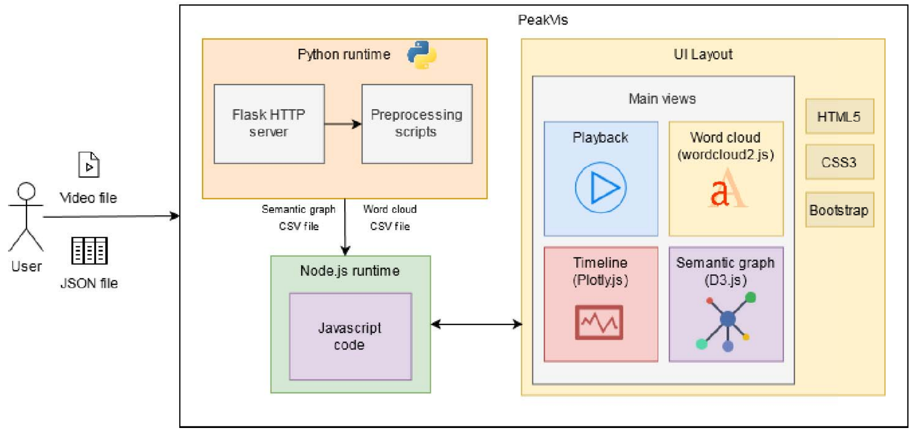

Fig. 2. PeakVis main components.

design, we use mainstay web technologies such as CSS3[5], Bootstrap[6], and HTML 5.

The tweet datasets used for analyzes in this paper were collected using the self-implemented TweetUtils toolkit[7], which uses the Twitter API[8] to crawl and retrieve tweets from the REST or real-time interfaces. However, TweetUtils is not required for operating PeakVis, given that other data collection utilities can be employed as long as the output JSON file matches the TweetUtils key-naming convention as detailed in the documentation.

Following the gathering of Tweets using keyword filters, the video and the tweet dataset can be loaded onto PeakVis. Upon loading a dataset through the interface, PeakVis will invoke a local Python script, which applies a few procedures to prepare the data for drawing the word cloud, and the semantic graph. Regarding the word cloud, we first perform basic cleaning of the text data, such as setting it to lowercase and stopword removal. The routines used to accomplish these tasks are derived from the TweetUtils package, allowing users to supply their own stopword list that may match their language of interest and packaged into one main script responsible for all of the preprocessing tasks. From the processed text data, we compute a key-value paired file counting the occurrences of every word in a 15-second interval, which is used then to draw and update the word cloud view every 15 seconds. This procedure generates a new comma-separated values (CSV) file, which is also loaded onto PeakVis to support the drawing of the word cloud.

Likewise, displaying the semantic graph also uses a supporting input file, which is also generated at the time of the dataset upload by the invoked Python script. We made use of the Python NetworkX library[9] for the semantic graph implementation. We first perform some text preprocessing in the tweets by going through the corpus of data while removing URLs, mentions, hashtags, and stopwords. Following this task, we generate a vocabulary V_{words} array from the remaining words after the preprocessing. This provides the basis for a graph data structure, a table with source and destination fields. The origin field starts from a permutation between the words of the sentence, where each word is repeated sequentially by the size of V_{words}. The destination field takes the subsequent words of origin. This procedure will yield a completely connected network for each tweet. The generated network is often too large for proper visualization; therefore, we filter the returning network to retrieve only the 20 most recurring words in the dataset. Finally, we run modularity algorithms from the library to group the network based on common connections. The display presentation was made with D3.js[10] with details on demand, indicating a node's closest connections on mouse hover. The generated graph is saved onto a CSV file and stored locally inside the PeakVis file structure. Given these preprocessing tasks can take a sizable amount of time according to dataset volume, PeakVis will check for the existence of the word cloud and the graph files before executing the aforementioned procedures, both of which should follow the naming convention of the uploaded tweet dataset. In the scenario in which they both exist, PeakVis will directly load these files, thus skipping the preprocessing-related tasks. Therefore, these operations only need to be completed once for a given dataset, as long as the uploaded file does not contain any changes in its file name.

Lastly, the Javascript version of Plotly[11] with Node.js bindings are used to draw the line chart and to integrate the wordcloud2.js[12] library to draw the word clouds. Any chart interactions in the line chart, such as zooming in and out in the time axis and seeking to segments in the video

[5]Cascading Style Sheets: https://www.w3.org/Style/CSS/
[6]Bootstrap: https://getbootstrap.com/
[7]Available at https://github.com/DAVINTLAB/TweetUtils
[8]Twitter API reference: https://developer.twitter.com/en/docs/api-reference-index
[9]NetworkX: https://networkx.github.io/
[10]D3js: https://d3js.org/
[11]Plotly: https://plotly.com/
[12]wordcloud2.js: https://wordcloud2-js.timdream.org

coordinately with the playback view, are handled by Plotly's own programming interface. Similarly, dynamically updating the responsive word cloud to the video playback time is performed through the wordcloud2.js programming interface.

C. Highlight detection method

The idea of the highlight detection method implemented is to identify all peaks in the input data and mark the peaks that satisfy the user-set highlight threshold condition. For this, we intuitively perceive the concept of "peaks" as a line chart segment that is demarcated by an ascending slope in the y axis and terminated by a descending slope. This is implemented by navigating the x–axis while comparing the volume of tweets of the current entry to the previous entry. If the current verified value is larger than the previous value, we mark this point as the start of an ascending slope. Again, a descending slope is marked whenever the current value in position is smaller than the previous value. We progressively perform this operation throughout the data while storing the starting position of the slope, maximum, and minimum values for each peak. However, performing this operation at very granular intervals may introduce problems such as prematurely closing a peak that has yet to develop. So, through empirical research, we identified that 25 seconds is a sliding interval that successfully works with Twitter data, smoothing out the curves and ensuring the stability of the method described.

After having each peak properly demarcated, we can now decide whether this peak is a highlight or not. We deemed that different users may have different points of view when looking for such foci of attention, e.g., different teams in a match, diverse contestants for a prize, best moves of referee-related issues, may result in varied highlights. Therefore, to emphasize the user interaction aspect and flexibly adapt to the users' needs, we introduced the highlight sensitivity threshold parameter. The highlight sensitivity threshold allows the user to specify the sensitivity in which PeakVis detects highlights. Then, if a low highlight sensitivity value is specified, only high peaks can be detected; instead, the higher the value, the smaller peaks can be detected. For example, when executing the tool on a soccer match, a low sensitivity threshold would detect only the most significant events in the game that had the most social media impact, such as goals from either team. A higher sensitivity threshold would also detect less significant events with smaller peaks, such as half-times and missed goal chances.

As shown in Algorithm 1, our highlight detection algorithm works by comparing the previously observed peaks to a quantile distribution. We first compute the cut points in a 10–quantile distribution in the input dataset. Next, we measure the difference between the maximum and lowest value for every peak marked in the previous step, notated by $\bar{u}$. For every $\bar{u}$ value, we verify on which quantile interval these values are situated. This is done by calling the `bisect` function, which returns the list index of the interval a value is placed at, given an arbitrary value – in this scenario, the $\bar{u}$ values, and an ordered list of values – the ordered D set of cut points for the 10-quantile distribution. The highlight sensitivity threshold is expressed in the interface as a slider, which is read as a value ranging from 1 to 9, corresponding to each cut point in the quantile distribution. Thus, for a given peak, if the difference between its highest and lowest point is larger than the n^{th} quantile represented by the sensitivity threshold, we mark this peak as a highlight.

Data: a P set of peaks containing the $P_{i_{(x,y)}}$ initial slope position, $P_{i_{\max}}$, and $P_{i_{\min}}$ values. An s highlight sensitivity threshold integer value ranging from 1 to 9. An ordered D set containing the cut points in a 10–quantile distribution for the input dataset.
Result: A set of highlight markers for every P_i that fits the highlight sensitivity threshold s condition.

for *each peak P_i in P* **do**
  $\bar{u} \leftarrow (P_{i_{\max}} - P_{i_{\min}})$
  $D_{idx} \leftarrow bisect(\bar{u}, D)$
  if $D_{idx} \geq s$ **then**
    mark $P_{i_{(x,y)}}$ as a highlight.
  end
end

Algorithm 1: Highlight detection algorithm.

IV. Analysis of PeakVis Applicability

To analyze the applicability of the proposed tool, we describe two case studies: in the first one, we analyzed the Big Brother Brazil reality show season finale from 2018; in the second, a Brazilian telenovela final episode from 2019. We chose to analyze these events because they drew large audiences to live broadcasts. Furthermore, we also analyzed the FIFA World Cup 2018 match between Brazil and Costa Rica. However, as expected, the key points were very easy to identify, such as the VAR-appointed penalty (34 minutes into the second half) and the two goals scored by the Brazilian team (in the last minutes). This fact supported the tool's highlight detection process. The other ten events highlighted by PeakVis had considerably less significance on the match than the primary three. Nevertheless, all were relevant moves as the moments when a goal almost happened, or a player with a subpar performance was replaced. The tweet frequency was proportional to the importance of the events in the field. Considering this, we decided to explore, here, only the two first mentioned studies.

To explore the possibilities of PeakVis in a broader sense, the authors invited two professionals from the Communication field to analyze the two case studies separately. They both graduated in Advertising and have experience with Digital Marketing. Reviewer n^{o}1 analyzed Big Brother Brazil 2018's season finale, using one's research expertise in fan culture and pop culture studies. His analysis sought to recognize the reasons why the tweet traffic peaks happened when compared to the reality show's narrative. Reviewer n^{o}2 analyzed the telenovela final episode, considering his previous experience in movie narratives and visual communication. This analysis aimed to identify which events correspond to the highlights. Given the different nature of both shows, we understood that the analysis should also fit each one's peculiarities and if the tool can perform on both research needs.

A. Data Collection

As mentioned in Subsection III-B, we collected tweets related to the two case studies by using the TweetUtils toolset.

The Big Brother Brazil 2018 finale aired on April 19^{th}, from which we retroactively collected tweets on April 22^{nd}. We specified the set of keywords according to the show's official hashtags as tweeted by the broadcaster's network official account, complementing with the name of the finalists: #BBB2018, #FinalBBB18, #BBB, Kaysar, and Gleici. The data collection amounted to a total of 1,838,508 tweets, including retweets. Regarding the telenovela, we collected the tweets of the final episode that took place on November 23^{rd}, 2019, also retroactively. According to the official broadcast terms, we specified #adonadopedaco and #adonadopedaço as keywords. This approach resulted in a total of 148,996 tweets, or 52,272 excluding retweets.

The two videos of the events were captured from live broadcasts with a DVD-R recorder. The broadcasts came from the local Globo networks' affiliate, the country's television ratings leader. The recordings were then exported and re-encoded with the video transcoder HandBrake.

B. Case Study 1: Big Brother Brazil 2018 Finale

The final episode of Big Brother Brazil 2018's reality show (from now on quoted as BBB) had a total length of 1 hour and thirty-five minutes and began at 10:37 PM on April 19, 2018. The sensitivity threshold was configured to show the most highlights possible, comprising 11 peaks. We briefly analyzed the first seven smaller peaks (represented in Figure 1) and did an in-depth description of the last three peaks that represent the higher tweet count among all of them.

One interesting fact about this BBB is that although there were four people in the final, one was a couple - composed by father and daughter (called Família Lima - Lima's family). The spectators had, then, three options to vote, even though there were four contestants: Gleici, Kaysar, and Família Lima (Ana Clara and Ayrton). The winner was Gleici, and the runner-up was Kaysar. The finale reached the highest audience ratings since the 2011 edition[13], indicating that the Brazilian viewers are still interested in this reality show franchise.

The first peaks were marked mostly by tweets commenting on situations that happened during the season. As the show made a retrospective of events and used popular television actors to deliver the remarks, PeakVis represented the Twitter users reacting to them. One pertinent point about those peaks are the relation to the varied "memes" created during the season, mocking and commenting on participant's attitudes. As the episode recapitulated those situations, tweeting peaks spiked, indicating the power of the fans in determining what is deemed to be relevant to the show's producers. Romance controversies between participants also stirred conversations on Twitter, creating new peaks as the show went by.

Something curious about our analysis of word clouds in these smaller peaks is that, although the most talked about subjects were related to parts of the program, in absolute numbers, the most present words are related to the favorite participants. The use of the hashtag #bbb18 and the words: "vote", "voted", "winning", "selling" were also recurring. Therefore, despite the fact that the program's events generate spikes in tweets, there is a constancy in the tweets in general, which are more focused on the fans for the possible winners. Therefore, although the subjects are relevant, they are not enough to appear in the word clouds as the most cited words.

The tool helped to remark the importance of creating an engaging narrative based on what the audience was talking about. More emotional scenes motivated some peaks, such as messages from the participants' families – we noticed that most supportive comments focused on the participant Kaysar, the Syrian immigrant. Even though he was not the favorite to win, he continued to be a conversation topic.

The peaks grow more prominent as the announcement over the winner was closing by. Peak 1 (Figure 1: 1) started at midnight when the host says he will announce the third place. There are a small peak and a larger one without a break, and the interval is short – before peak 1, the third place was announced. A few minutes later, at 12:02 AM, came the announcement of Gleici as the winner. A significant increase in the number of posts follows this. There is a recurrence of participant support posts celebrating her victory, many referring to her as "fairy", her nickname on the show. The participants left the house, partying, and there is a decline in tweets just after the peak of celebrations.

Peak 2 (Figure 1: 2) begins at 00:05 AM when the host starts talking to participants after they celebrate with their families and with the other colleagues in the program. At the time of filming, seemingly, Gleici speaks "Lula Livre," and this becomes a trigger for Twitter users to comment on the subject.

It is interesting to note that, in this case, there is an overlap of the pure-entertainment BBB program and a political event. "Lula Livre" refers to the support of voters from Luís Inácio Lula da Silva, a former Brazilian president who had been arrested at the time. It is not possible to know for sure if this was the phrase uttered by Gleici, but in the recordings' audio track, it is possible to understand something similar. The public then reacts because it is an unusual and unexpected fact.

The last peak, Figure 1: 3, occurs at 00:08 AM. The comments refer to the moment when the host talks to Kaysar, saying that they are in contact with the United Nations and trying to solve the situation of his family in Syria. The speech was brief, but enough for Twitter users to comment on the outcome, as exemplified in these two tweets (translated to English and contextually adapted for better understanding): (1) "They are going to bring Kaysar's family. That's beautiful.";

[13] https://f5.folha.uol.com.br/televisao/bbb18/2018/04/final-do-bbb-18-teve-maior-audiencia-desde-2011.shtml

(2) "#BBB18 THEY ARE BRINGING KAYSAR'S FAMILY I'M SO EMOTIONAL OH MY GOD". The number of tweets began to decrease by the end of the show.

We also compared the word clouds of the three highest peaks, as shown in Figure 3, where each one considered the words from the first tweet until the respective peak. This analysis showed us something similar to what happened in the smaller peaks. It is impossible to detect which specific subject is discussed at the peak from the analysis of the datasets. Therefore, the word clouds help in the general semantic perception, but they cannot translate the perception we had when looking at the tweets in detail. It is possible to notice the prevalence of the same words in the three different peaks, revealing that it would not be possible to draw the previous conclusions without a specific look in the moment's most retweeted tweets.

Fig. 3. Word clouds of the three highest peaks for BBB 2018 finale.

Furthermore, analyzing all the 11 peaks, we noticed some patterns in the BBB 2018. A sharp increase in user feedback was expected during the final winner announcement. Still, it was curious to note that even after this moment, there is another peak related to Kaysar, the runner-up. There were two peak moments related to this participant and his family situation in Syria, and a situation regularly addressed in the program. We realize that moments that spark emotion can mobilize the audience. The peak that relates to Gleici's speech to the former president also brings us essential information about the audience and how political issues also mobilize users, even in entertainment contexts.

Another analysis used the semantic graph with the 1000 most retweeted tweets. The graph represents the conversation throughout the program, and it was interesting to observe which terms had more connections in common. This tool helped us to understand users' general feelings about the main participants: Gleici and Kaysar (see Figure 4). Figure 4a shows that the words that have the most common connections are: "final", "#bbb18", "#finalbbb18", "best"(melhor), "edition"(edição) and "Lula". In addition to the hashtags referring to the program, two subjects attracted a lot of comments: this would be the best edition of the program according to viewers/Twitter users, and the repercussion of the sentence uttered by Gleici when leaving confinement ("Free Lula"). In the Kaysar graph (Figure 4b), the words with the most common connections are: "family"(família), "UN"(onu), "Brazil"(brasil) and "#BBBnetwork"('#redebbb). Kaysar's peaks shown the repercussion of his refugee status and his struggle to reunite with his family in the background. What is reflected in the semantic graph is precisely this, the mobilization of both Globo network and Twitter users so that the participant's situation had a positive outcome. Therefore, the semantic graphs represent more comprehensively the specific analysis made at each peak and give legitimacy to the retweets of the three peaks analyzed in detail.

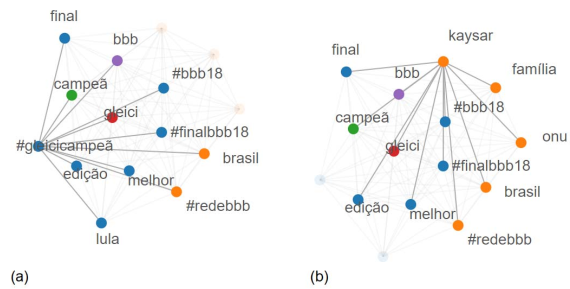

Fig. 4. Semantic graph for the most retweeted tweets.

C. *Case Study 2: "A Dona do Pedaço" telenovela*

Telenovelas (a television soap opera in Latin America) are the top primetime fictional product of Brazilian television. Alencar [19] and Hamburger [20] argue that their capacity to engage audiences during daily broadcasts lasting months depends on their plots and characters to motivate viewers' debates. Our validation prototype presented here was the last episode of the telenovela "A Dona do Pedaço" (The Boss of the Block), which ran from May to November 2019, and aired its final on Friday, November 22.

When analyzing the Twitter database, it was possible to establish some inferences. Thus, motivated by a preliminary analysis, not discussed here, we only considered the original tweets for this analysis (not the retweets ones). The narrative highlighted the struggle of rival families Ramirez and Matheus through the decades, led by the character Maria da Paz, who earned her livelihood by making cakes. Through ups and downs, Maria's adventures with her family and their foes developed seven main story arcs and gripped spectators in a constant flow of comments on the internet.

The telenovela's final episode was expected to bring closure to this main story arcs and their characters. Although this goal was achieved, the Peakvis tool highlighted that the internet conversation proved to be much more selective. Only two of the seven stories generated significant interest online, with the eponymous Maria da Paz being largely forgotten.

The most recurrent words in the messages after filtering the search terms were: Josiane, Final, Novela (telenovela in Portuguese), Chiclete, Camilo, and Vivi. In these words, it is possible to identify that the main highlight was among the four characters Chiclete, Camilo, Vivi, and Josiane. The first three belong to a story arc in which the digital influencer Vivi was confined by Camilo, a corrupt police officer, and ends up being rescued in a violent scene by her true love, Chiclete, a hitman. Josiane starred in the final scene in the last arc. Using the PeakVis tool to analyze the last episode and its respective database, it was possible to establish some notes. When adjusting the tool's sensitivity to the intermediate level, we identified five peaks of tweets (highlights) along with the transmission, shown in Figure 5.

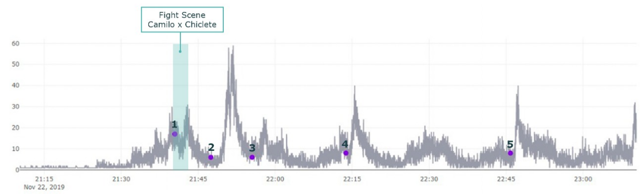

Fig. 5. Visualization of total Tweets in the last episode with highlights of the peaks.

It was possible to observe that the most prominent peak, between points 2 and 3, occurred during the exhibition of the scene in which the influencer Vivi Guedes using her smartphone, said goodbye to her followers and her career. However, what was more prominently discussed in the tweets is related to an earlier scene: the fight between Chiclete and Camilo (which results in Camilo's death). As highlighted by Figure 5, this scene took place approximately 10 minutes before the peak, but its trail of discussion spread over other segments and story arcs. The repercussion included messages such as (translated to English and contextually adapted for better understanding) "Finally, this unfortunate died #ADonaDoPedaço" and "Chiclete as always carrying the telenovela on his back ♡ #ADonaDoPedaço", which indicates the satisfaction of the audience with the unfolding of this plot.

The tweets near peak 3, already in the telenovela interval, continued to address the fight scene. Only after this peak did the publications on the farewell scene by Vivi Guedes began to grow in parallel.

The word clouds analysis in Figure 6 shows that the highest frequency during the transmission of the episode were the names of the characters involved in the fight at the beginning of the episode: Camilo and Chiclete. This aspect indicates that the fight that led to Camilo's murder and the release of Vivi from her unhappy marriage dominated the conversation until the final scene. The multiple presentations of the same word, such as "chiclete", "Chiclete", and "CHICLETE" suggest the different spellings used by online viewers.

The analysis suggests a decline in interest in the outcomes of the other arcs from the last episode. Accompanying the peaks' reduction, a more significant number of tweets criticizing the telenovela began to grow as the broadcast progressed, complaining about uninteresting and predictable conclusions. The final scene with the character Josiane changed this feeling, giving audiences a surprise (although predictable) and generating half of the collection traffic after display, analyzed outside the PeakVis tool, and therefore, outside of this paper's scope.

Fig. 6. Word clouds of the six highest peaks for the telenovela's final episode.

V. Final Remarks

Considering that more and more people tend to watch television broadcasts and at the same time use social media, it is important to analyze them simultaneously. Thus, in this work, we presented PeakVis, an interactive tool that syncs a video recording with tweets, allowing the analysis of broadcast highlights identified through Twitter traffic peaks and the visualization of the top messages, semantic graphs, and word clouds. Our main contributions are the proposal of a novel interactive approach to identify these broadcast highlights based on social media and the evidence of its possibilities of use through two case studies: the last episodes of a reality show and a telenovela. Both cases feature a Brazilian content bias originating from the authors' cultural background and professional locus.

Compared with related work, PeakVis was designed for different events, allowing the specification of the sensitivity to detect highlight and syncing between quantitative and semantic data and video, also enabling human-centered contextual commentary. Thus, regarding the semantic features of PeakVis, in the first case study, the word clouds brought results convergent with the other visualizations. Despite that, we believe that there is little evidence to say that this feature is not useful in the tool since the second case study allows us to observe that one of the story arcs dominated almost the entire episode of the telenovela.

We chose one case study with competition focused on opposing fans. This polarization leaves little room for ambiguity on the postings, hampering to detect details by spikes in the word clouds. On the other hand, the word clouds helped to identify the moment of the telenovela episode that had more repercussions. Also, in the case of BBB, the semantic graphs reinforce in a certain way the qualitative analysis of the tweets of each peak, as there is the presence of the words mentioned in specific tweets. Thus, we can have a quantitative confirmation of the previous conclusions. The graphs can then indicate a general feeling and bring up the relationship between different expressions used by Twitter users.

The trend towards the smartphone's simultaneous use with television needs to be observed by media producers, as they indicate an activity behavior about the show. Conversations on social media are commonplace, and PeakVis showed great potential for second-screen media analysis wherever such broadcast and online habits converge. It has proven useful for entertainment and reality shows, as it can show more nuances of behavior that go beyond the most outstanding moments, such as announcing winners, for example.

Among the possibilities of the market use of PeakVis, is the understanding of the public's reaction to different situations in audiovisual productions, especially on television (open or cable), but not limited to it, and that it can help analyze any broadcast or live-streamed event. Advertisers can also take the opportunity to understand user behavior toward potential peaks of interest by directing ads to times when they are most interested in receiving messages. They can also organize publication times according to past histories of interest in specific program blocks. Similarly, broadcasters can establish different advertisement insertion values based on proven narrative moments with the highest engagement potential. It is also possible to develop a video highlights cut based on public interest, possibly making it automatic for social media publishing or aiding professional video editing software through an EDL (Edit Decision List) integration. Besides that, it is possible to understand which narrative events of the shows (in this case, specifically reality shows) draw the most public attention on Twitter. With that information, producers can create shows that can engage even more audience attention and generate more conversation online.

We believe that there is potential for future analysis and consequently future improvements of PeakVis considering other kinds of programs or entertainment productions, such as politics and talk shows. Analyzing them, it will be possible to understand other possibilities for using word clouds and the semantic graph, for instance. As next steps, we intend to enable its use in real-time. We are already allowing the use of a YouTube link, but we have to address other issues as the processing of a huge number of tweets and the possibility of removing words that maybe hinder the clarity of the analysis.

ACKNOWLEDGMENT

This work was partially supported by PUCRS (Chamada Apoio a Integração entre Áreas - PRAIAS 5/2020). There was also support from UolEdTech. Isabel H. Manssour also would like to thank the financial support of the CNPq Scholarship - Brazil (308456/2020-3).

REFERENCES

[1] S. Harrington, T. Highfield, and A. Bruns, "More than a backchannel: Twitter and television," *Participations*, vol. 10, no. 1, pp. 405–409, 2013.

[2] T. D. Baruah, "Effectiveness of social media as a tool of communication and its potential for technology enabled connections: A micro-level study," *International Journal of Scientific and Research Publications*, vol. 2, no. 5, pp. 1–10, 2012.

[3] H. Harb and L. Chen, "Highlights detection in sports videos based on audio analysis," in *Proceedings of the 3rd International Workshop on Content-Based Multimedia Indexing*, 2003, pp. 22–24.

[4] A. Hanjalic, "Adaptive extraction of highlights from a sport video based on excitement modeling," *IEEE Transactions on Multimedia*, vol. 7, no. 6, pp. 1114–1122, Dec 2005.

[5] J. Lanagan and A. F. Smeaton, "Using twitter to detect and tag important events in sports media," in *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*. Barcelona, Spain: AAAI Press, 2011, pp. 542—-545.

[6] A. Tang and S. Boring, "#Epicplay: Crowd-sourcing sports video highlights," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '12. New York, NY, USA: ACM, 2012, pp. 1569–1572. [Online]. Available: http://doi.acm.org/10.1145/2207676.2208622

[7] S. Jai-Andaloussi, A. Mohamed, N. Madrane, and A. Sekkaki, "Soccer video summarization using video content analysis and social media streams," in *Proceedings of the 2014 IEEE/ACM International Symposium on Big Data Computing*, ser. BDC '14. USA: IEEE Computer Society, 2014, p. 1–7. [Online]. Available: https://doi.org/10.1109/BDC.2014.20

[8] K. Doman, T. Tomita, I. Ide, D. Deguchi, and H. Murase, "Event detection based on twitter enthusiasm degree for generating a sports highlight video," in *Proceedings of the 22Nd ACM International Conference on Multimedia*, ser. MM '14. New York, NY, USA: ACM, 2014, pp. 949–952. [Online]. Available: http://doi.acm.org/10.1145/2647868.2654973

[9] G. Fião, T. Romão, N. Correia, P. Centieiro, and A. E. Dias, "Automatic generation of sport video highlights based on fan's emotions and content," in *Proceedings of the 13th International Conference on Advances in Computer Entertainment Technology*, ser. ACE '16. New York, NY, USA: Association for Computing Machinery, 2016. [Online]. Available: https://doi.org/10.1145/3001773.3001802

[10] T. Romão, P. Centieiro, G. Fião, N. Correia, and A. E. Dias, "Designing a system for the automatic generation of sport video summaries," in *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, ser. EICS '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 69–74. [Online]. Available: https://doi.org/10.1145/3102113.3102130

[11] C. Koleejan, H. Takamura, and M. Okumura, "Generating objective summaries of sports matches using social media," in *2019 IEEE/WIC/ACM International Conference on Web Intelligence*. IEEE, 2019, pp. 353–357.

[12] S. Zhao, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "Human as real-time sensors of social and physical events: A case study of twitter and sports games," *CoRR*, vol. abs/1106.4300, pp. 1114–1122, 2011. [Online]. Available: http://arxiv.org/abs/1106.4300

[13] L.-C. Hsieh, C.-W. Lee, T.-H. Chiu, and W. Hsu, "Live semantic sport highlight detection based on analyzing tweets of twitter," in *2012 IEEE International Conference on Multimedia and Expo*. Melbourne, VIC, Australia: IEEE, July 2012, pp. 949–954.

[14] G. Oorschot, M. Erp, and C. Dijkshoorn, "Automatic extraction of soccer game events from twitter," in *Proceedings of the Workhop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2012)*. Boston, USA: CEUR-WS.org, 2012, pp. 21–30. [Online]. Available: http://ceur-ws.org/Vol-902/paper_3.pdf

[15] Y. Huang, C. Shen, and T. Li, "Event summarization for sports games using twitter streams," *World Wide Web*, vol. 21, no. 3, pp. 609–627, 2018.

[16] J. Hannon, K. McCarthy, J. Lynch, and B. Smyth, "Personalized and automatic social summarization of events in video," in *Proceedings of the 16th International Conference on Intelligent User Interfaces*, ser. IUI '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 335–338. [Online]. Available: https://doi.org/10.1145/1943403.1943459

[17] M. Nakazawa, M. Erdmann, K. Hoashi, and C. Ono, "Social indexing of tv programs: Detection and labeling of significant tv scenes by twitter analysis," in *Proceedings of the 26th International Conference on Advanced Information Networking and Applications Workshops*. USA: IEEE Computer Society, 2012, pp. 141–146.

[18] T. Hayama, "Detecting tv program highlight scenes using twitter data classified by twitter user behavior and evaluating it to soccer game tv programs," *IEICE Transactions on Information and Systems*, vol. E101.D, no. 4, pp. 917–924, 2018.

[19] M. Alencar, *A Hollywood brasileira: panorama da telenovela no Brasil*. São Paulo: SENAC, 2002.

[20] E. Hamburger, *O Brasil antenado: a sociedade da novela*. Rio de Janeiro: Zahar, 2005.