Using Oliver API for emotion-aware movie content characterization

Theodoros Giannakopoulos, Spiros Dimopoulos, Georgios Pantazopoulos, Aggelina Chatziagapi,
Dimitris Sgouropoulos, Athanasios Katsamanis, Alexandros Potamianos, Shrikanth Narayanan
Behavioral Signal Technologies Inc
Los Angeles, CA, USA
Email: {thodoris, sdim, pantaz, aggelina, dimitris, nassos, alex, shri}@behavioralsignals.com

***Abstract*—This paper demonstrates the utilization of *Oliver* [1], the speech emotion recognition (SER) API created by Behavioral Signals, in the context of a movie content visualization application. Oliver API provides an emotion recognition as-a-service solution that can be accessed via a Web API. In this work, we demonstrate how one can send sound recordings from famous movies, retrieve respective emotional descriptors and use simple aggregations on these descriptors to visualize movie content. We have compiled a dataset of 60 movies, categorized over 8 directors. The classification examples included in this paper indicate the ability of simple emotion aggregations to discriminate between movie directors. In order for others to also experiment with the output of both the API's Emotional and Automatic Speech Recognition, the responses are provided as JSON files in this link: https://tinyurl.com/yxeqvvy2.**

***Index Terms*—speech emotion recognition; API; behavioral analysis; automatic speech recognition; content-based characterization**

I. Introduction

Content-based indexing methodologies have been helping us manage the huge amounts of data available online of various types and modalities, by utilizing intelligent content search and recommendation. A popular application of these methodologies is movie recommendation systems, which are either based on user preferences (*collaborative systems*) or movie attributes that are statistically mapped to the user preferences (*content-based systems*) or a combination of both (*hybrid systems*). Towards this direction, several research efforts have occurred during the last years [1], [2]. Currently, state-of-the-art systems that provide movie recommendation services, are either collaborative systems, such as *MovieLens*[2], or content-based systems, like *jinni*[3], or hybrid systems, as is *IMDb*[4].

Content-based systems typically rely on human-generated metadata without taking into account the raw multimodal content of the movie itself, i.e. audio, visual and textual channels of the movie. To utilize the content of a movie, automatically extracting high-level content descriptors from movies' multimodal signal is necessary for enhancing the content-based indexing process. As far as the audio modality is concerned, in [3] deep convolutional neural networks are used to predict latent factors from music audio signals and apply them in music recommendation. Additionally, a model designed for violent content detection has also been proposed [4]. In [5], a video recommendation system is based on stylistic visual features. An extension of the previous system takes advantage of visual cues from trailers, as well as human-generated tags [6], [7]. With regards to the textual domain, in [8] they constructed a hierarchical neural network generative model for movie dialogues. In [9] a movie topic discriminator from text is also investigated. More advanced approaches combine the nature of the modalities. A mixture of audio-visual features is adopted for movie genre classification in [10] and emotion recognition in [11], [12]. Other studies focus on audio-visual representations of particular aspects of the movie such as speaker gender [13] and speakers clustering [14]. Finally, a work combining all the available modalities is presented in [15] aiming to project the raw content of the movie directly to a higher level representation.

High-level descriptors have also been combined with collaborative knowledge in hybrid systems. In [16] they suggest an innovative approach in which a hybrid recommendation system is binding the collaborative knowledge from social movie networks with the high-level representation originated from topic models. Another hybrid recommendation system is presented in [17], in which the authors focus on the fusion of collaborative filtering and sentiment classification of movie reviews to boost the final results.

In this paper, we demonstrate how automatically generated estimates of emotions from speech can be used to provide representations and enhance content-based visualization of movies. Toward this end, we have used Oliver, our emotion API that extracts emotions and behaviors from audio streams. By combining simple aggregates on the Oliver's results, along with the application of simple dimensionality reduction algorithms, we illustrate how content representation can also be based on emotional content from the movies. In this way, we demonstrate how emotion can be used as an additive ”dimension” when visualizing and indexing movie content and how is this dimension correlated to particular movie attributes such as the movie director or the movie's genre.

[1] https://behavioralsignals.com/oliver/

[2] https://movielens.org/

[3] http://www.jinni.com/

[4] http://www.imdb.com/

978-1-7281-4673-7/19/$31.00 ©2019 IEEE

II. Oliver API

Oliver is Behavioral Signal's Emotion Artificial Intelligence (EAI) API. Developers can directly benefit from Oliver's growing emotional intelligence, measure emotions and behaviors in conversations, and utilize our continuously evolving robust analytics in their own applications. Either that involves development of a virtual assistant (VA) for a business, an interactive game for children, a voice-controlled speaker for the home, or a social robot designated to assist the elderly, incorporating emotion-aware spoken language understanding will supercharge your users experience.

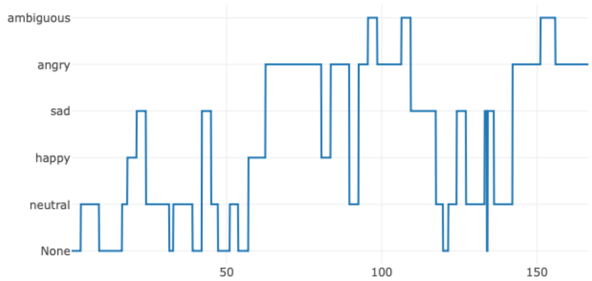

Fig. 1. Frame-level emotions sequence. Y axis denotes the behavior output of the API whereas X axis denotes time measured in secs. To read the JSON frame-level results of the API one can use the sample code available at https://github.com/behavioral-signals/best-api-analyze/. The video analyzed in this example is this https://www.youtube.com/watch?v=pRE23YfSvc8 from the Pulp Fiction movie.

Oliver API provides an emotion recognition as a service solution that can be accessed via a Web API. The client can use the REST API to submit audio and/or video files in batch mode, poll for the status of processing and get the results of the emotion estimates when ready. The input can be a URL pointing to an audio/video file in the web or the actual audio file content in multipart html form content type uploaded from the client's system. In general, the API endpoint responses are provided in JSON format. The *Process* endpoint contains information about the progress of processing and can be polled to discover when the processing is complete. The *Results* endpoint comprises of audio-based analysis. In particular, it returns frame-level emotion recognition predictions and signal-level analysis such as diarization or KPI results which are aggregated in *Events*. Events are computed based on speaker identity information, so that each event corresponds to a speaker turn, a variable size audio segment containing speech originated from a single speaker. The *Results-frames* endpoint returns frame-level analysis results with a fixed step of 200 msec. Meanwhile, the user can access that particular endpoint to create custom aggregations. A streaming API using web sockets is also provided as an alternative mode of operation to the batch API. The streaming API can be utilized for any application that requires fast setup of processing and low latency of response.

To aid a prospective client to start using the API, a CLI tool[5] is provided to interact with the API. The CLI is simple to use and as a bare minimum it requires the client to prepare a csv file containing the audio files local filesystem path and their respective number of channels. In addition, we have made publicly available open source clients of the streaming API written in Python[6] and Javascript[7] which can be readily used to bootstrap a possible use case scenario.

As described above, emotion estimates are provided either in a fix-sized frame-level resolution, i.e. one decision every short-term frame, or in a event-level, i.e. in the form of aggregated results. In this demo, we are focusing only on the ”emotion” attribute of the API, which includes the following emotional labels: non-speech (this includes either non-speech classes or silence), neutral, angry, sad, happy and ambiguous. The ambiguous label indicates speech in which the system cannot discern a specific emotional state. This is automatically extracted based on the confidences of the classification pipeline. A visualization of the behavior outputs of the Oliver API is illustrated in Figure 1, in which the available emotions are displayed in contrast with time. To read the frame-level API responses one can also use the open-source sample code available here https://github.com/behavioral-signals/best-api-analyze/.

III. Use case data

In order to demonstrate the ability of Oliver API to extract emotions from speech signals in movies, that can be used to discriminate between different movie content, we have selected several movies from eight famous movie directors. The director names and the number of movies per director in the dataset are presented in Table I. The total number of movies is 60 and the dataset is not fully balanced (per director). In addition, for reducing demands on computational resources, we have selected not to send the audio stream that corresponds to the whole movie but the signal after the first hour of audio data has been removed (i.e. almost second half of the movie for an average movie duration). The two types of API responses used in this demo (ASR and emotions) are shared as JSON files here [8]. The format of these files has been can be found at http://oliver.readme.io.

IV. Content Representation and Visualization

A. Emotional representation

In this demo, we leverage the frame-level emotion estimates of the Oliver API. In particular, we send the audio streams of the corresponding movies use case dataset and gather the API responses which consist of a list of emotion estimates for each file. Therefore, each audio frame has been automatically labelled by Oliver API in one of the following 6 audio classes,

[5] https://bitbucket.org/behavioralsignals/api-cli

[6] https://bitbucket.org/behavioralsignals/python-streaming-client/src

[7] https://bitbucket.org/behavioralsignals/js-streaming-client/src/master/

[8] https://tinyurl.com/yxeqvvy2

TABLE I
DATASET DESCRIPTION

Director Name	movies in dataset
Darren Aronofsky	5
Coen brothers	7
Francis Ford Coppola	5
Roman Polanski	6
Martin Scorsese	9
Stanley Kubrick	5
Quentin Tarantino	8
Woody Allen	15

as mentioned above: non-speech, neutral, angry, sad, happy and ambiguous.

As soon as the sequences of emotion estimates are retrieved for each audio movie recording, we proceed with a simple aggregation of these emotions. Specifically, we calculate the percentages of each emotional class (including non-speech) in the whole movie. This leads to *a 6-dimensional feature vector, one percentage for each audio class*. This simple representation aims to demonstrate the ability of the API to produce "emotional signatures" for audio streams. Obviously, other, more sophisticated approaches can be used to aggregate emotional estimates, but this is beyond the purpose of this demo paper.

B. Text-based representation

In addition to the speech emotions estimated by the API for each input audio stream, we have also selected to use the ASR output (described above), to achieve text-based content representation. Towards this end, we parse the text returned from the JSON ASR and apply GloVe embeddings [18], a widely adopted unsupervised learning method used to extract vector representations at word level. Training in GloVe has been performed on aggregated global word-word co-occurrence statistics from a corpus. In particular, for demo purposes we have selected to use the 50-dimensional GloVe representation trained on the Wikipedia dataset (6B) [9]. These embeddings are used to create one vector per word. Finally, the generated words are averaged to form a fixed text representation accounting for the file as a whole.

C. 2-D content visualization

To visualize the movie content in 2-D space, we chose to apply Principal component analysis (PCA) as a dimensionality reduction method from the two aforementioned feature representations (the 6-D speech emotion aggregates and the 50-D text-based representations) to the 1-D dimension for each initial representation method. In addition, we have selected to train a basic Support Vector Machine classifier on the final 2-D representation space, with director names used as ground truth labels, to illustrate the "decision surfaces" between the individual movie directors.

V. RESULTS

Figures 2 and 3 illustrate the 2-dimensional emotion and textual content for the movies of the directors. For visualization purposes, we decided to divide the directors into two groups which are represented in each figure separately. Moreover, in order to evaluate the ability of the textual and emotional features to discriminate between directors we have performed a simple k-fold validation on the initial "feature spaces" (50-d for text and 6-d for emotion). Our results showed that the textual features can achieve up to 40% normalized f1-score, while the emotional representation achieves 24% (baseline random prediction is 13%).

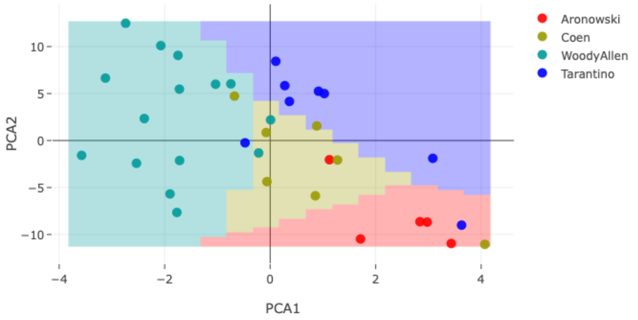

Fig. 2. Emotion and textual content distribution for movies of the directors: Aronofsky, Coen brothers, Woody Allen and Tarantino

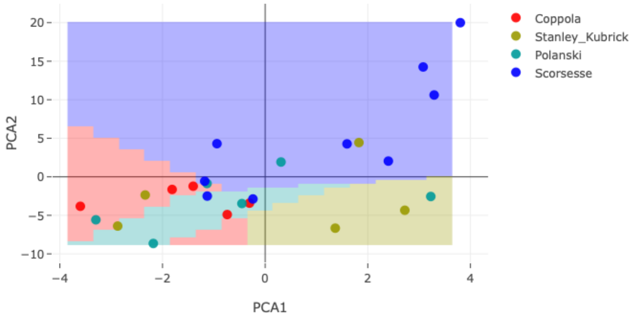

Fig. 3. Emotion and textual content distribution for movies of the directors: Coppola, Kubrick, Polanski and Scorsese.

Some qualitative notes:

- Coen brothers and Roman Polanski are illustrated as the most "compact" directors in terms of both dimensions.
- Movies from Aronofsky and Scorsese are most often "outliers", considering both text and emotion.
- Woody Allen can be distinguished from Tarantino, Coen brothers and Aronofsky by only using the emotional

[9] https://nlp.stanford.edu/projects/glove/

representation, i.e. the horizontal PCA axis, with accuracy that reaches almost 100%.

VI. Conclusions

In this paper we have demonstrated the utilization of frame-level speech emotion recognition results, produced by Oliver API, in the context of a movie content characterization pipeline. Visualizations and classification performance measures have indicated the ability of the aggregated emotional features, combined with textual descriptors of the ASR outputs (also extracted by Oliver API), to discriminate between movie directors. The overall vision of adopting this emotion-based content representation is to discover latent knowledge essential to correlate emotional content situated in movies and users' preferences.

In the future, we will demonstrate how this emotional information can be used in a real-world movie recommendation system that results in emotion-aware explanatory predictions. Additionally, we will work on more sophisticated emotion aggregation methods that also take into account the dynamics of the dialogs and the temporal evolution of emotions.

References

[1] B. N. Miller, I. Albert, S. K. Lam, J. A. Konstan, and J. Riedl, "Movielens unplugged: experiences with an occasionally connected recommender system," in *Proceedings of the 8th International Conference on Intelligent User Interfaces*. ACM, 2003, pp. 263–266.

[2] Z. Wang, X. Yu, N. Feng, and Z. Wang, "An improved collaborative movie recommendation system using computational intelligence," *Journal of Visual Languages & Computing*, vol. 25, no. 6, pp. 667–675, 2014.

[3] A. Van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Advances in Neural Information Processing Systems*, 2013, pp. 2643–2651.

[4] T. Giannakopoulos, D. Kosmopoulos, A. Aristidou, and S. Theodoridis, "Violence content classification using audio features," in *Proceedings of the 4th Helenic Conference on Advances in Artificial Intelligence*, ser. SETN'06. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 502–507.

[5] Y. Deldjoo, M. Elahi, P. Cremonesi, F. Garzotto, P. Piazzolla, and M. Quadrana, "Content-based video recommendation system based on stylistic visual features," *Journal on Data Semantics*, pp. 1–15, 2016.

[6] Y. Deldjoo, M. Elahi, P. Cremonesi, F. B. Moghaddam, and A. L. E. Caielli, *How to Combine Visual Features with Tags to Improve Movie Recommendation Accuracy?* Springer International Publishing, 2017, pp. 34–45.

[7] Y. Deldjoo, M. Quadrana, M. Elahi, and P. Cremonesi, "Using mise-en-scène visual features based on MPEG-7 and deep learning for movie recommendation," *ArXiv preprint*, vol. arXiv:1704.06109, pp. 1–8, 2017.

[8] I. V. Serban, A. Sordoni, Y. Bengio, A. C. Courville, and J. Pineau, "Hierarchical neural network generative models for movie dialogues," 2015.

[9] C. Dupuy, F. Bach, and C. Diot, "Qualitative and descriptive topic extraction from movie reviews using lda," in *International Conference on Machine Learning and Data Mining in Pattern Recognition*. Springer, 2017, pp. 91–106.

[10] Z. Rasheed and M. Shah, "Movie genre classification by exploiting audio-visual features of previews," in *Proceedings of the 16th International Conference on Pattern Recognition, 2002.*, vol. 2. IEEE, 2002, pp. 1086–1089.

[11] N. Malandrakis, A. Potamianos, G. Evangelopoulos, and A. Zlatintsi, "A supervised approach to movie emotion tracking." in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 2376–2379.

[12] S. E. Kahou, C. Pal, X. Bouthillier, P. Froumenty, c. Gülçehre, R. Memisevic, P. Vincent, A. Courville, Y. Bengio, R. C. Ferrari, M. Mirza, S. Jean, P.-L. Carrier, Y. Dauphin, N. Boulanger-Lewandowski, A. Aggarwal, J. Zumer, P. Lamblin, J.-P. Raymond, G. Desjardins, R. Pascanu, D. Warde-Farley, A. Torabi, A. Sharma, E. Bengio, M. Côté, K. R. Konda, and Z. Wu, "Combining modality specific deep neural networks for emotion recognition in video," in *Proceedings of the 15th ACM on International Conference on Multimodal Interaction*, ser. ICMI '13. New York, NY, USA: ACM, 2013, pp. 543–550.

[13] T. Guha, C.-W. Huang, N. Kumar, Y. Zhu, and S. S. Narayanan, "Gender representation in cinematic content: A multimodal approach," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ser. ICMI '15. New York, NY, USA: ACM, 2015, pp. 31–34.

[14] I. Kapsouras, A. Tefas, N. Nikolaidis, G. Peeters, L. Benaroya, and I. Pitas, "Multimodal speaker clustering in full length movies," *Multimedia Tools and Applications*, vol. 76, no. 2, pp. 2223–2242, Jan 2017.

[15] K. Bougiatiotis and T. Giannakopoulos, "Enhanced movie content similarity based on textual, auditory and visual information," *Expert Systems with Applications*, vol. 96, pp. 86–102, 2018.

[16] S. Wei, X. Zheng, D. Chen, and C. Chen, "A hybrid approach for movie recommendation via tags and ratings," *Electronic Commerce Research and Applications*, vol. 18, pp. 83–94, 2016.

[17] V. Singh, M. Mukherjee, and G. Mehta, "Combining collaborative filtering and sentiment classification for improved movie recommendations," *Multi-disciplinary Trends in Artificial Intelligence*, pp. 38–50, 2011.

[18] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.