I know how you feel now, and here's why!: Demystifying time-continuous high resolution text-based affect predictions in the wild

Vedhas Pandit, Maximilian Schmitt, Nicholas Cummins, Bjorn Schuller

Angaben zur Veröffentlichung / Publication details:

Pandit, Vedhas, Maximilian Schmitt, Nicholas Cummins, and Bjorn Schuller. 2019. "I know how you feel now, and here's why!: Demystifying time-continuous high resolution text-based affect predictions in the wild." *32nd International Symposium on Computer-Based Medical Systems (CBMS), 5-7 June 2019, Cordoba, Spain*, 465–70. https://doi.org/10.1109/cbms.2019.00096.

I *know* how you feel *now*, and here's *why*!: Demystifying Time-continuous High Resolution Text-based Affect Predictions In the Wild

Vedhas Pandit[1], Maximilian Schmitt[1], Nicholas Cummins[1], Björn Schuller[1,2]
[1] *ZD.B Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Germany*
[2] *GLAM – Group on Language, Audio & Music, Imperial College London, UK*
{vedhas.pandit, schuller}@informatik.uni-augsburg.de

***Abstract*—Affective computing 'in the wild' is of huge relevance to the healthcare field, like it is for many industries today. Applications of direct relevance are patient monitoring (e. g., emotional state, depression and pain monitoring), health information mining, diagnosis and opinion mining (e. g., from medical reports and drug reviews). The prevalence of the text modality in the medical field for various reasons – e. g., privacy laws, high costs and prohibitory memory requirements for audio and video data – has made the text modality the most popular. Deviating away from traditionally a classification task at a sample-level, the promising baseline results for the Audio/Visual Emotion Challenge (AVEC) 2017 make a strong case for the suitability of text data for a 'time-continuous' affect estimation. For the very first time, we present insights into the inner workings of a *deep learning, 'in the wild' affect-predicting, time-continuous regression model*. We compute relevance of the sparse text-based bag-of-words features (BoTW) of the AVEC 2017 challenge in estimating the three affect labels, viz. arousal, valence and liking, by using a layerwise relevance propagation method (LRP). Interestingly, the trained models are found to rely more on adjectives and adverbs such as 'schlecht', 'gut', 'genau' with positive or negative connotations, and action descriptors such as <laughter> and <slightlaughter> – quite analogous to the human perception of emotion expression.**

***Keywords*-Affective Computing, AVEC, Bag-of-Words, Feature Relevance, In the Wild, Input-times Gradient, Layerwise Relevance Propagation.**

I. Introduction

'Text-based sentiment analysis' has become a popular research direction in recent years, thanks to ubiquity of opinion-rich text resources (e. g., medical and official records, public databases, social media), rapid advancements in natural language processing and machine learning, surge in open-source frameworks, and the ever-growing industry demand. These developments have far reaching implications and applications in the healthcare field likewise [1, 2].

As an example, emotional intelligence (EI) of the doctors is known to be directly related to their job satisfaction, stress management, and their trustability [3–5]. Because EI is a strong objective predictor for a patient-doctor relationship, and as a consequence, the patient's response to a treatment [6, 7], the need for EI coaching of doctors and nurses has long been conclusively established [8, 9]. In addition to pedagogy, emotionally aware human computer interaction (HCI) is also a promising assistive technology for autistic patients to help identify and express emotions [10, 11]. Affective computing research will likely revolutionise real-time patient monitoring, e. g., through remote pain, depression and emotional state monitoring, and personal health information mining [12, 13]. Text-based sentiment analysis is useful for accelerated understanding of the drug and treatment reviews [14], assessing certainty of a diagnosis from a medical report [1], surveying patient sentiment for medical services [15], and to understand underlying relationships between neurophysiological signals and emotions [16, 17].

A. Motivation

When building any emotion-aware application, we face predominantly four challenges: 1. a model's ability to work on a real-life data (robustness) 2. introducing time-continuous prediction capabilities, 3. introducing understanding of affect niceties (affect resolution), 4. having interpretable models.

It is essential that the trained application is robust enough to work on in the wild data, i. e., data recorded under non-laboratory settings. The training and test data is expected to have noise, missing values, errors, misalignments, or inaccurate timing information. For introducing time-continuous prediction capabilities, we require not a single, but a sequence of affect labels during training. The labels should ideally be high-resolution value-continuous labels, and not merely the classification labels. Additionally, even for an exceptionally-performing model, it is desirable that we are able to gain insights into inner workings of the model and make sure that the model is interpreting the data the way it is meant to be. Interpretable models also help better our (i. e., human) understanding of the problem and the solution, saving us from making costly errors [18].

B. SEWA/AVEC 2017 Corpus

The 'Automatic Sentiment Analysis in the Wild' (SEWA) corpus [19] is the *only* in the wild public database available to date, featuring time-continuous, high resolution labels for multiple dimensions of affect. The success of bag-of-words (BoW)-based text features of the SEWA database ('Affect Recognition' sub-challenge of the Audio/Visual Emotion Challenge and Workshops (AVEC 2017) [20]) makes this database an interesting use-case for understanding AI-based time-continuous multi-dimensional emotion prediction.

C. Organisation of the Paper

In Section II and Section III, we discuss in depth the superior performance and the extraction of textual features in the context of AVEC 2017. Section IV explains the key concepts of layer-wise relevance propagation (LRP) method using which we have computed the relevance scores for the individual features. We present our key findings in Section VI. Lastly, we conclude with a scope for future work in Section VII.

Table I: AVEC'17 baseline performance with SVRs using the BoW textual features alone, C= Complexity [20].

Data Split	Emotion	C	CCC	PCC	RMSE
Devel	Arousal	2^{-6}	0.3713	0.4591	0.1365
	Valence	2^{-7}	0.3907	0.4840	0.1379
	Liking	1	0.3147	0.3289	0.1240
Test	Arousal	2^{-6}	0.3775	0.4223	0.1036
	Valence	2^{-7}	0.4245	0.4945	0.1050
	Liking	1	0.2462	0.2832	0.1551

II. The AVEC 2017 Database

The SEWA corpus features subject pairs conversing in one of the six languages, i. e., Chinese, English, German, Greek, Hungarian, and Serbian. The subjects were asked to discuss a quite repetitive water-tap commercial that was otherwise informative, and generally likeable. We use a subset of the SEWA corpus used previously in the 'Affect Recognition' sub-challenge of the AVEC 2017 [20], featuring German subjects. The subject pairs dominate the conversations mostly equally in most of the cases. The mean duration of conversations is 2 minutes 47 seconds. The data was split into training, development and test set in roughly 2:1:1 ratio [20].

III. The BoW Textual Features of AVEC 2017

A. Baseline Results and Suitability of the Features

While the video, audio and textual features were provided to the challenge participants, the baseline paper reported effectiveness of the text features across all of the three affect dimensions, including the challenging 'liking' dimension (cf. Table I). The participants, too, used the text features invariably – despite the availability of much context-rich, highly informational video and audio data – many even reporting enhancements in the prediction upon incrementally using the text features [20–23]. The information contained in the textual features is, thus, highly informative and non-redundant.

As the very purpose of this study is to investigate how even a simple feedforward neural network-based regression model is able to exploit the information contained in the BoW textual features to reasonably predict the affect dimensions continuously in time, without the need of any post-processing step, we discuss the feature extraction process in-depth, including those details missing from the AVEC 2017 baseline paper due to space constraint [20].

B. Feature Extraction Algorithm In Depth

The database originally consisted of only the orthographic transcriptions labelled by a native German speaker. These labels were mostly at a sentence level, but may consist of even a single word or an action marker (e. g., laughter), depending on how the annotator chose to divide the speech to transcribe it (cf. directory: *transcription*). As a preprocessing step, every word in these transcriptions was roughly time-aligned through a simple linear interpolation, using the relative position of a word in comparison to the total number of words in an utterance, and the start- and end-times of the utterance. Every word in a labelled utterance, thus, corresponds to a unique timestamp; timestamp that is rounded to some integer step of 10 ms. Because no 'forced alignment' tool was used, it is obvious that this timing information is likely highly inaccurate (cf. directory: *textual_features*). Next, a vocabulary is built with only the words with at least two occurrences in the training partition. A bag-of-text-words (BoTW) representation is then computed using histograms of frequencies of occurrence of words from the vocabulary, for a moving window of 6 seconds (hop=0.1 second). Because the vocabulary consists of 521 words, and because we use only unigrams, the BoTW features consist of 521 dimensions, where the feature value itself represents logarithm of the term frequency [20] (cf. Figure 1).

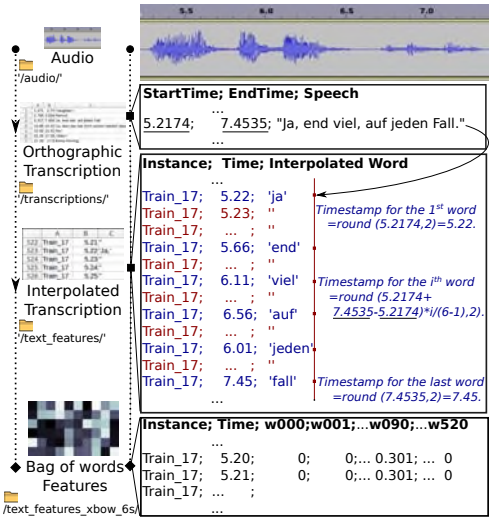

Figure 1: Complete feature generation process pipeline. The approximate word locations are first obtained using interpolation in time. The rest of the timestamps are marked with an empty string. The term frequencies (tf_n) for 521 words in a 6 second window are computed (hop=100 ms), and feature values are then assigned as ($w_n = log_{10}(tf_n + 1)$ if $tf_n \geq 1$ else 0 $\forall n \in [0, 520]$). In the interest of reproducibility[1], the corresponding directory names are given above.

C. Key Characteristics of AVEC 2017 BoTW Features

To better understand likely reasons for the success of BoTW features, It is important to revisit and establish first what this histogram-based, sparse, yet effective feature representation achieves, what kind of knowledge it encapsulates.

1) The BoW feature transformation, when applied on a moving window, captures the varying temporal trends in the distribution of quantisations of the input feature vectors (cf. Figure 2). Therefore, while this is also a downsampling operation, it better preserves a certain degree of proximal context. The representation provides a better summarisation of the adjacencies, and the temporal trends in the original feature space (cf. Figure 2).

[1]Code available at http://github.com/vedhasua/ExplainAVEC

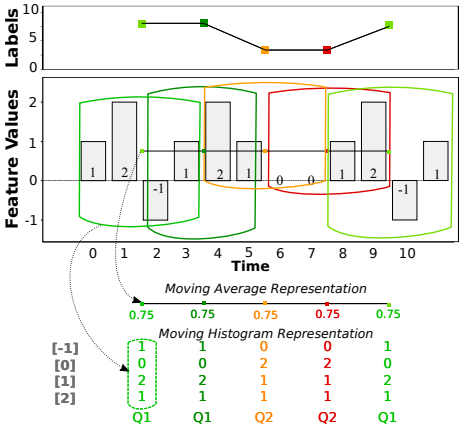

Figure 2: A statistical summarisation (e. g., mean, median) over a fixed-width window counteracts effects of outliers in the original feature space, yet capturing temporal dynamics. In the toy example above, however, a 4-point moving average (hop size=2) results in a critical loss of information. The down-sampled feature sequence is constant valued, which cannot be mapped to dynamically varying output labels. However, note that the moving histogram-based representation ([Q1,Q1,Q2,Q2,Q1]) preserves more of the context. It correctly captures the fact that the the 3rd and 4th histograms – featuring 0-valued features that drive the label value down to 2.5 from 7.5 – differ vastly from the rest of the histograms. BoW-based downsampling, thus, is more context and adjacency information-preserving.

2) The low level descriptors (LLDs) are typically computed over a small data frame, and may feature outliers. The moving BoW representation is inherently less sensitive to the outlier LLDs, compared to the moving average for example, due to the intermediate quantisation step and the inherent sparsity [24, 25].
3) When the moving BoW features are computed over largely overlapping frames, the BoW transformations of consecutive frames are closely inter-related. This is because the changes in the corresponding histograms are then gradual, and consequently, so are the changes expected in the corresponding outputs for a generally continuous input-to-output mapping. Moving BoW representation is, therefore, ideal to model a gradually varying sequential data – i. e., the regression labels.
4) As for the challenge feature set, every feature value represents not only the occurrence or non-occurrence of a certain word, but a the logarithm of the term frequency in the surrou ng 6 second window.
5) No complex pretrained eddings were necessary to generate the provided n eric feature transformation from the textual space t e numerical space.
6) BoW feature vector wit ll zeros (indicating absence of all of the words and t action markers) corresponds to a neutral affective sta in theory, i. e., label = 0.0, for arousal, valence and lik g.

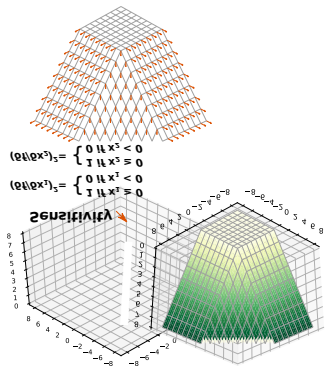

Figure 3: Why the LRP/DTD-based relevance computations are necessary: The sensitivity score $\equiv (\frac{df}{dx_i})^2$ quantifies the effect of the inputs on the output *when changed*. The LRP/DTD-based relevance score $\equiv$ *contributions* of the individual inputs in generating the output. Above, the sensitivity scores incorrectly imply that x_1 and x_2 are equally relevant to the output y at $(41, 1)$ and $(1, 41)$; both scores = 1. Obviously however, x_1 contributes the most in raising the output y to 42 at $(41, 1)$. The relevance score $[f(x)]_i$ defined as $max(0, x_i)$ makes more sense in this case. The LRP/DTD methods achieve that.

IV. Feature Relevance Computation

A popular technique to quantify relevance of inputs is to compute the effect of change in inputs on the output; i. e., the more the effect, the more relevant is the input. However, this *sensitivity* score may not reflect the true relevance of inputs, since (1) partial derivatives are indicative of the local effects, and (2) there is no direct relationship between the value of a function and the partial derivative at a data-point [26]. For example, in Figure 3, both inputs are considered equally relevant as per the sensitivity score when $x_1, x_2 \geq 0$, $\because \left(\frac{dy}{dx_1}\right)^2 = \left(\frac{dy}{dx_2}\right)^2 = 1$. However, we know that the contributions to y from x_1 and x_2 vary in the (x_1, x_2) space. Further, this metric is discontinuous at $x_1 = 0$ and at $x_2 = 0$.

The 'Deep Taylor Decomposition' (DTD) method [27] for relevance computation is a special case of LRP [26, 28]. Both the LRP and DTD methods aim to provide constituent contributions coming from individual features for an output. For the example given in Figure 3, the LRP and DTD-based relevance functions effectively capture different contributions coming from x_1 and x_2, while remaining continuous at $x_1 = 0$ and $x_2 = 0$. Intuitively speaking, when applied to an image recognition model, sensitivity tells us 'change in which pixels' would make the image less or more of a dog image, while a DTD/LRP-based decomposition tells us the extent to which each pixel contributes to make it a dog image.

A. LRP formulation used in this paper

LRP-Z, a basic building block of LRP, implements a conservative decomposition of relevance score ($R_j^{(l)}$) in proportion to $z_{ij} = x_i w_{ij}$ signal sent from a neuron i in the l^{th} layer to a neuron j in the $(l+1)^{th}$ layer in the forward pass [26], where

w_{ij} is a weight mapping input x_i to output x_j, with bias b_j.

$$R_{i\leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j} \cdot R_j^{(l+1)} \text{ where } z_j = \sum_j z_{ij} + b_j, x_j = g(z_j) \quad (1)$$

In Equation (1), g is an activation function, which may be non-linear. When a network consists of only the linear, max-pooling and rectified linear units layers, $R_j^{(l)}$ for any input x_i equals the value of the input multiplied by partial derivative of the output with respect to that input, or $R_i = x_i \cdot \frac{\partial f(x)}{\partial x_i}$.

Note that for the given non-negative, sparse BoTW feature representation, DeepLIFT (another contribution-oriented relevance score computation) [29] reduces to $R_i = x_i \cdot \frac{\partial f(x)}{\partial x_i}$ likewise. This is primarily because the 0-valued input vector maps to a 0-valued regression label (cf. Section III-C.6). Thus, while introducing non-linearity to the model through rectified linear units, contribution-based relevance computations remain easy to compute, to debug, and are readily interpretable.

V. Training of the Regression Models

A. The Model Topologies and Training

Because a neural network learns with random weight initialisations, different training iterations give rise to completely differently trained models. For any given topology of a model, we run the training multiple times until a certain number of best generalising, different models are generated – exceeding baseline performance on both the development and test data split. To help the model learn the property Section III-C.6, we augment the training data by about 50 % new training samples featuring 0-valued input feature vectors and the corresponding 0-valued output regression outputs for the three affect dimensions. To help avoid overfitting and aid model generalisation with more of fault tolerance, we inject small random gaussian noise (mean=0, variance=2.5e-6) into to our training partition [30]. Overfitting was additionally avoided using regularisers and dropouts.

The models were trained using the Keras framework with Tensorflow [31] as the backend. We experimented with multiple combinations of the number of hidden layers ($\in$ *[0,3]*) and the number of nodes for the hidden layers ($\in 2^{[3,4,\cdots,8]}$), and optimisers ($\in$ *{adamax, adagrad, adam, rmsprop}*). We use the concordance correlation coefficient (CCC) as the performance metric, as in AVEC 2017. We used a novel loss function to train our model, presented next.

B. Novel MSE and CCC-based Loss Function

The many-to-many mapping between the typical loss function mean square error (MSE), and the AVEC 2017 performance metric (CCC) is given by the following equation [32].

$$CCC = \left(1 + \frac{MSE}{2 \cdot \sigma_{XY}}\right)^{-1},$$

$$\text{where, } \sigma_{XY} = \text{Covariance}(\text{prediction}, \text{gold-standard}) \quad (2)$$

We use $(\frac{MSE}{\sigma_{XY}})^2$ as the loss function for quicker training of our models, as we want to optimise for a high CCC. The use of square of the term $(\frac{MSE}{\sigma_{XY}})$ is to ensure that the network does not 'cheat', i. e., minimise the loss function without optimising for MSE, but rather by merely making the covariance more negative. The squared $(\frac{MSE}{\sigma_{XY}})$ formulation ensures that MSE gets minimised, while square of the covariance is simultaneously maximised. Expectedly, as a side effect, we do run into situations where the trained model results in a highly negative CCC on the training and validation data. However, as discussed in Section V-A, we analyse only the well-trained models as dictated by the performance thresholds of the AVEC 2017 baseline results.

VI. Results

The goal of this paper is to gain insights into the inner working of reasonably trained models, learning from their reasonably accurate predictions. To this end, we compute feature relevance scores for from the test set samples, where the prediction error is less than the mean absolute error. We use the iNNvestigate toolkit [33] to compute the relevances. We plot the most positively and most negatively contributing features, capturing the temporal dynamics of the relevance scores against the input values (cf. Figure 4). Note that even these 'most relevant' features seldom contribute to the output; owing to the sparse nature of the inputs.

While the insights we discuss next are consistent across different network topologies, Figure 4 was generated using the network topology of 3 hidden layers with [256, 64, 16] nodes, and [linear, relu, relu] activations respectively. Because relu activation inhibits the output negation, we avoid using it for the output layer in our analysis. Our output layer consists of one node with a linear activation. Likewise, the linear activation at the input side gives the model an opportunity to vary the negative weights (thus, relevances) across the BoTW features. The nodes of the first hidden layer, thus, generate a derived (or a secondary) feature-set consisting of linear combinaions of the input BoTW features. The next 2 hidden layers introduce non-linearity in the input to output mapping, thanks to the relu activation.

Indeed, as a human would, the arousal and valence dimensions are heavily influenced by the laughter-related action markers. The model typically finds the small words such as 'und' (and), 'super' (super/great), 'nicht' (not), 'auch' (also/too) particularly useful in predicting high arousal regression outputs. These words were used often for collaborative interruption (e. g., and/also), or when exclaiming (e. g., super), or to emphasise (e. g., too/also). The action markers for fillers (<filler>), conjunctions such as 'also' (contextually 'well..'/so/hence in English), 'irgendwie' (somehow), 'vielleicht' (perhaps/maybe) have a negative effect on the arousal prediction. We note that these words indicate a certain degree of uncertainty, hesitation, an act of thinking. When used in a conversation, these words were often coupled with short pauses in speech, decreasing arousal level.

Because the relavance score for the <laughter> was drastically high, the effect of other words is expected to manifest mostly in its absence. The plot of prediction errors, BoTW input values for <laughter> ($= log(tf_{<laughter>} + 1)$, and relevance scores for other BoTW features confirm our hypothesis (available in the source code). Further, the arousal, valence

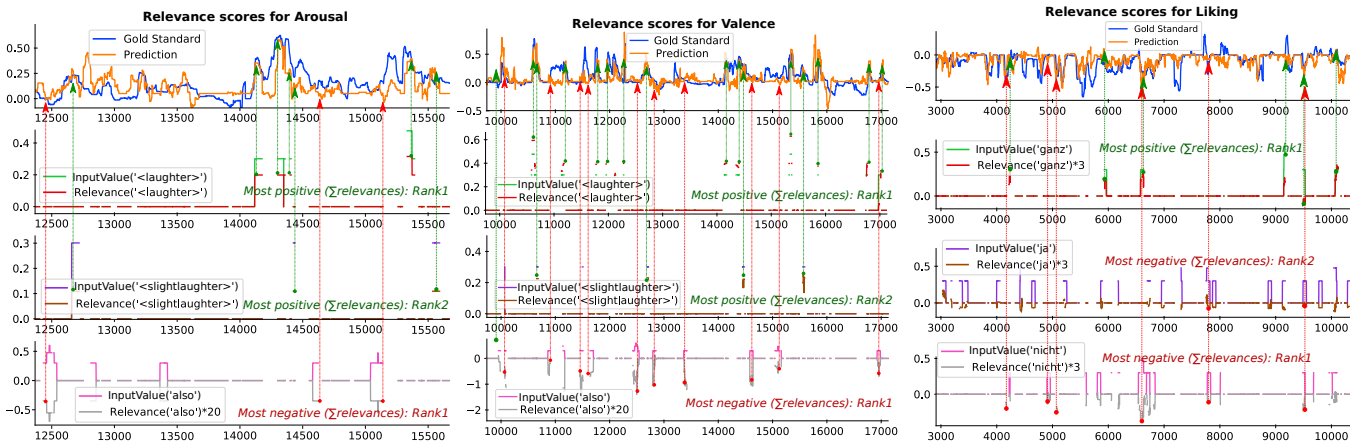

Figure 4: For a trained model, illustrations above capture temporal dynamics of the feature values and the relevance scores in predicting three affect dimensions, where the prediction error was less than a certain threshold. For arousal and valence dimensions, the crests and the troughs are very highly correlated with the most positively and negatively contributing features respectively. Predictions for liking do not exhibit as much a temporal correlation. The highly positively and negatively contributing words are found to be mostly consistent with human perception of the emotion expression.

peaks correlate well with the most negatively and positively contributing features (cf. Figure 4).

Interestingly, when predicting the most challenging 'Liking' dimension, the model focuses more on words, rather than the laughter-related markers. It also often attends to affirmative words such as 'ja' (yes/yeah), 'echt' (really), adjectives like 'schon' (nice/already), 'gut' (good), 'groß' (size/big), 'ganze' (complete/very). People often talk about the reasons why they (dis)liked something. Interestingly, the liking model assigns high relevance to words implying contexts e. g., 'dazu'/'außerdem' (therefore), 'endlich' (at last), 'über' (over (something)), 'zusammenhang' (context), 'weil' (because). However, the model assigns a high relevance to only a few context-indicating words invariably (e. g., at last/because/thus) and the choice of these words changes in every training iteration. Thus, average scores for these words across different models get diminished. These words were not considered as relevant in predicting arousal and valence.

The liking dimension also exposes a few more limitations of the models and our approach – owing to low-performance of the 'liking' prediction model. Because the CCC threshold is low = 0.246 (unlike e. g., 0.378 for arousal), the liking dimension has a higher mean absolute error; and we end up computing feature relevances across more erroneous predictions. Consequently, the troughs and valleys in the time-domain do not correlate as highly with the most positively and the most negatively contributing feature values for 'liking', unlike the other two dimensions. Likewise, the German definite article 'die' is often considered highly relevant, inconsistent with human perception. This might be because of its prevalence in the German texts (since 'die' indicates feminine singulars, and all plurals), and because the relevance score is directly proportional to the feature value. Consistent with human perception however, 'Wasserhahn'-like nouns (water-tap in English) were observed to be irrelevant across all emotion dimensions.

VII. Conclusions and Future Work

For the very first time, we *investigate* and *reason* the success of BoTW features in predicting time-continuous, high resolution, multidimensional affect on in the wild data, using contribution-based relevance score computation. The models, for the most part, were observed to utilise the features consistent with human perception of emotion expression.

We come to understand a few limitations of the models, thanks to the challenging liking dimension where the model predictions were inferior (although higher than the challenge baseline). We aim to address these limitations by merging together certain BoTW features (e. g., the articles, 'because' with 'hence'). While we present a pioneering attempt at reasoning performance of a neural network on time-continuous, in the wild, multidimensional affect regression, all the conversations have a consistent theme. We intend to run similar experiments on more of in the wild data with varied themes for the conversations. It would be of interest to demystify more advanced architectures (e. g., attention-based recurrent and convolutional neural networks), and also relevance of different dimensions of the word-embeddings in time-continuous affect prediction.

VIII. Acknowledgements

This research was supported by the EU's Horizon 2020 Programme through the Innovative Action No. 826506 (susTAGE). We thank the developers of 'iNNvestigate', especially S. Lapuschkin and M. Alber for their prompt responses to our queries, and for implementing the suggested tool fixes.

References

[1] K. Denecke and Y. Deng. Sentiment analysis in medical settings: New opportunities and challenges. *Artificial Intelligence in Medicine*, 64(1):17–27, 2015.

[2] R. W. Picard and J. Healey. Affective wearables. *Personal Technologies*, 1(4):231–240, 1997.

[3] H.-C. Weng, C.-M. Hung, Y.-T. Liu, et al. Associations between emotional intelligence and doctor burnout, job satisfaction and patient satisfaction. *Medical Education*, 45(8):835–842, 2011.

[4] S. Arora, H. Ashrafian, R. Davis, et al. Emotional intelligence in medicine: a systematic review through the context of the acgme competencies. *Medical Education*, 44(8):749–764, 2010.

[5] A. Finset and T. A. Mjaaland. The medical consultation viewed as a value chain: a neurobehavioral approach to emotion regulation in doctor–patient interaction. *Patient Education and Counseling*, 74(3):323–330, 2009.

[6] F. Benedetti. Placebo and the new physiology of the doctor-patient relationship. *Physiological Reviews*, 93(3):1207–1246, 2013.

[7] R. L. Street Jr, G. Makoul, N. K. Arora, et al. How does communication heal? pathways linking clinician–patient communication to health outcomes. *Patient Education and Counseling*, 74(3):295–301, 2009.

[8] A. C. McQueen. Emotional intelligence in nursing work. *Journal of Advanced Nursing*, 47(1):101–108, 2004.

[9] H.-C. Weng, H.-C. Chen, H.-J. Chen, et al. Doctors' emotional intelligence and the patient–doctor relationship. *Medical Education*, 42(7):703–711, 2008.

[10] O. Rudovic, J. Lee, M. Dai, et al. Personalized machine learning for robot perception of affect and engagement in autism therapy. *Science Robotics*, 3(19), June 2018.

[11] R. W. Picard, E. Vyzas, and J. Healey. Toward machine emotional intelligence: Analysis of affective physiological state. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(10):1175–1191, 2001.

[12] C. Lisetti, F. Nasoz, C. LeRouge, et al. Developing multimodal intelligent affective interfaces for tele-home health care. *International Journal of Human-Computer Studies*, 59(1-2):245–255, 2003.

[13] R. Kocielnik, N. Sidorova, F. M. Maggi, et al. Smart technologies for long-term stress monitoring at work. In *26th Intl. Symp. Computer-Based Medical Systems (CBMS)*, pages 53–58. IEEE, 2013.

[14] G. K. Savova, J. J. Masanz, P. V. Ogren, et al. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513, 2010.

[15] F. Greaves, D. Ramirez-Cano, C. Millett, et al. Harnessing the cloud of patient experience: using social media to detect poor quality healthcare. *BMJ Qual Saf*, 22(3):251–255, 2013.

[16] C. A. Frantzidis, C. Bratsas, C. L. Papadelis, et al. Toward emotion aware computing: an integrated approach using multichannel neurophysiological recordings and affective visual stimuli. *IEEE Transactions on Information Technology in Biomedicine*, 14(3):589–597, 2010.

[17] H. Kurniawan, A. V. Maslov, and M. Pechenizkiy. Stress detection from speech and galvanic skin response signals. In *26th Intl. Symp. Computer-Based Medical Systems (CBMS)*. IEEE, 2013.

[18] Z. C. Lipton. The mythos of model interpretability. *arXiv preprint:1606.03490*, 2016.

[19] J. Kossaifi, R. Walecki, Y. Panagakis, et al. SEWA DB: A Rich Database for Audio-Visual Emotion and Sentiment Research in the Wild. *arXiv preprint:1901.02839*, 2019.

[20] F. Ringeval, B. Schuller, M. Valstar, et al. AVEC 2017 – Real-life Depression, and Affect Recognition Workshop and Challenge. In *Proc. 7th Int. Workshop on Audio/Visual Emotion Challenge (AVEC'17) at 25th ACM MM*, pages 3–9, Mountain View, CA, Oct. 2017. ACM.

[21] J. Huang, Y. Li, J. Tao, et al. Continuous multimodal emotion prediction based on long short term memory recurrent neural network. In *Proc. 7th Int. Workshop on Audio/Visual Emotion Challenge (AVEC'17) at 25th ACM MM*, pages 11–18. ACM, 2017.

[22] S. Chen, Q. Jin, J. Zhao, et al. Multimodal multi-task learning for dimensional and continuous emotion recognition. In *Proceedings of the 7th Annual Workshop on Audio/Visual Emotion Challenge*, pages 19–26. ACM, 2017.

[23] T. Dang, B. Stasak, Z. Huang, et al. Investigating word affect features and fusion of probabilistic predictions incorporating uncertainty in avec 2017. In *Proc. 7th Int. Workshop on Audio/Visual Emotion Challenge (AVEC'17) at 25th ACM MM*, pages 27–35. ACM, 2017.

[24] V. Pandit, N. Cummins, M. Schmitt, et al. Tracking Authentic and In-the-wild Emotions using Speech. In *Proc. 1st ACII Asia 2018*, Beijing, P. R. China, May 2018. AAAC, IEEE.

[25] V. Pandit, M. Schmitt, N. Cummins, et al. How Good Is Your Model 'Really'? On 'Wildness' of the In-the-wild Speech-based Affect Recognisers. In *Proc. 20th Intl. Conf. Speech and Computer, SPECOM 2018*, Leipzig, Germany, Sep. 2018. ISCA, Springer.

[26] S. Bach, A. Binder, G. Montavon, et al. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PloS One*, 10(7):e0130140, 2015.

[27] G. Montavon, S. Lapuschkin, A. Binder, et al. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017.

[28] A. Binder, S. Bach, G. Montavon, et al. Layer-wise relevance propagation for deep neural network architectures. In *Information Science and Applications (ICISA) 2016*, pages 913–922. Springer, 2016.

[29] A. Shrikumar, P. Greenside, and A. Kundaje. Learning important features through propagating activation differences. *arXiv preprint:1704.02685*, 2017.

[30] C. M. Bishop. Training with noise is equivalent to tikhonov regularization. *Neural computation*, 7(1):108–116, 1995.

[31] M. Abadi, P. Barham, J. Chen, et al. TensorFlow: A System for Large-Scale Machine Learning. In *OSDI*, volume 16, pages 265–283, 2016.

[32] V. Pandit and B. Schuller. On Many-To-Many Mappings Between Concordance Correlation Coefficient and Mean Square Error. *arXiv preprint:1902.05180*, 2019.

[33] M. Alber, S. Lapuschkin, P. Seegerer, et al. iNNvestigate neural networks! *arXiv preprint:1808.04260*, 2018.