

Will Affective Computing Emerge from Foundation Models and General AI? A First Evaluation on ChatGPT

Mostafa M. Amin

University of Augsburg; SyncPilot GmbH

Erik Cambria

Nanyang Technological University

Björn W. Schuller

University of Augsburg; Imperial College London

Abstract—ChatGPT has shown the potential of emerging general artificial intelligence capabilities, as it has demonstrated competent performance across many natural language processing tasks. In this work, we evaluate the capabilities of ChatGPT to perform text classification on three affective computing problems, namely, big-five personality prediction, sentiment analysis, and suicide tendency detection. We utilise three baselines, a robust language model (RoBERTa-base), a legacy word model with pretrained embeddings (Word2Vec), and a simple bag-of-words baseline (BoW). Results show that the RoBERTa trained for a specific downstream task generally has a superior performance. On the other hand, ChatGPT provides decent results, and is relatively comparable to the Word2Vec and BoW baselines. ChatGPT further shows robustness against noisy data, where Word2Vec models achieve worse results due to noise. Results indicate that ChatGPT is a good generalist model that is capable of achieving good results across various problems without any specialised training, however, it is not as good as a specialised model for a downstream task.

■ **WITH THE ADVENT** of increasingly large-data trained general purpose machine learning models, a new era of ‘foundation models’ has started. According to [1], these are marked by

having been trained on ‘broad’ data – often self-supervised – at scale leading to a) homogenisation (i. e., most use the same model for fine-tuning and training for down-stream tasks as they are effective

across many tasks and too cost-intensive to train individually) and b) emergence (i. e., tasks can be solved that these models were not originally trained upon – potentially even without additional fine-tuning or downstream training). However, at this time, much more research is needed to understand the actual emergence abilities that potentially lead to a massive shift of paradigm in machine learning. Models might not need to be trained any more at all specifically for limited tasks, be it from the upstream or downstream perspective. Here, we consider the example of Affective Computing tasks seen from a Natural Language Processing (NLP) end. In the future, will we need to train extra models at all to tackle tasks such as personality, sentiment, or suicidal tendency recognition from text? Or will ‘big’ foundation models suffice with their emergence of these?

To this end, we consider ChatGPT as our basis for ‘big’ foundation model to check for the full emergence of the above three tasks. It was launched on 30th of November 2022, and had gained over one million users within one week [2]. It has shown very promising results as an interactive chatting bot that is capable to a big extent to understand questions posed by humans, and give meaningful answers to. ChatGPT is one of the above named ‘foundation models’ constructed by fine-tuning a Large Language Model (LLM), namely GPT-3, which can generate English text. The model is fine-tuned using Reinforcement Learning from Human Feedback (RLHF) [3], which makes use of reward models that rank responses based on different criteria; the reward models are then used to sample a more general space of responses [3], [4]. As a result, general artificial intelligence capabilities emerged from this training mechanism, which resulted in a very fast adoption of ChatGPT by many users in a very short time [2]. The effectiveness of these capabilities is not exactly known yet, for example, [5] explored many of the systematic failures of ChatGPT. [6] explains the history of development of Natural Language Processing (NLP) literature until arriving at the point of developing ChatGPT.

In summary, the aim of this paper is to systematise an evaluation framework for evaluating the performance of ChatGPT on various classification tasks to answer the question whether it shows full emergence features of other (Affective Computing-

related) NLP tasks. We use this framework to show if ChatGPT has general capabilities that could yield competent performance on affective computing problems. The evaluation compares against specialised models that are specifically trained on the downstream tasks. The contributions of this paper are:

- We evaluate, whether (NLP) foundation models can lead to ‘full’ (i. e., no need for fine-tuning or downstream training) emergence of other tasks, which would usually be trained on specific data sources; therefore,
- We introduce a method to evaluate ChatGPT on classification tasks.
- We compare the results of ChatGPT on three classification problems in the field of affective computing. The problems are big-five personality prediction, sentiment analysis, and suicide and depression detection.

The remainder of this paper is organised as follows: We begin by elaborating on the related work, then we introduce our method, then we present and discuss the results. We finish with concluding remarks.

RELATED WORK

We focus on related work within the key research question of potential emergence (in the text domain) by foundation models. In particular, [7] explores the question if ChatGPT is a general NLP solver that works for all problems. They explore a wide range of tasks, like reasoning, text summarisation, named entity recognition, and sentiment analysis. [8] explores the capabilities of GPT language models (including ChatGPT) in Machine Translation. [5] explores the systematic errors of ChatGPT.

METHOD

The aim of this paper is to evaluate the generalisation capabilities of ChatGPT across a wide range of affective computing tasks. In order to assess this, we utilise three datasets corresponding to three different problems, as mentioned: big-five personality prediction, sentiment analysis, and suicide tendency assessment. For these tasks, we utilise three datasets. On each of the introduced tasks, we attempt to get ChatGPT’s assessment about each of the examples of the corresponding Test set. Furthermore, we compare ChatGPT

Dataset	Train	Dev	Test	+ve	-ve
O				333	176
C				286	223
E	6,000	2,000	509	214	295
A				340	169
N				274	235
Sent	1,440,144	159,856	359	182	177
Sui	138,479	6,270	496	165	331

Table 1. Statistics on the sizes of the datasets, with counts of positive and negative classes in the Test set. The Test column shows the final number of samples used for evaluation (lower than the original sizes due to the limitation of manually collecting examples from ChatGPT).

against three baselines, namely a large language model, a word model with pre-trained embeddings, a basic bag-of-words model without making use of any external data. We describe the datasets, querying procedure of ChatGPT, and the baselines in this section. Figure 1 demonstrates the pipelines of all methods (ChatGPT and the three baselines).

Datasets

We introduce the three datasets in this Section. A summary of their statistics is presented in Table 1. We utilise publicly available datasets for reproducibility.

Personality Dataset We utilise the First Impressions (FI) dataset [9], [10] for the personality task¹. Personality is represented by the Big-five personality traits (OCEAN), namely, *Openness to experience*, *Conscientiousness*, *Extraversion*, *Agreeableness*, and *Neuroticism*. The dataset consists of 15 seconds videos with one speaker, whose personality was manually labelled. Such labelling was conducted by relative comparisons between pairs of videos, by ranking which person scores higher on each one of the big-five personality traits. A statistical model was then used to reduce the labels into regression values within the range [0, 1]. The personality labels were given based on the multiple modalities of a video, namely, images, audio, and text (content). We utilise the transcriptions of these videos as the input to be used to predict personality from. We use the Train/Dev/Test split given by the publishers of the dataset [9], [10]. Like [11], we train the models on this dataset as a regression problem (by using Mean Absolute Error as loss function), since the

¹We acquired the dataset on 03.02.2023 from <https://chalearnlap.cvc.uab.cat/dataset/24/description/>

labels can give a more granular estimation of the labels; then, we binarise these to positive or negative using the threshold 0.5.

Sentiment Dataset We adopt the Sentiment140 dataset [12] for the sentiment analysis task². The dataset is collected from tweets on Twitter, which makes the text very noisy, which can pose a challenge against many models (especially word models). The dataset consists of tweets and the corresponding sentiment labels (positive, or negative). We split the training portion with a ratio 9:1 to give the Train and Dev portions³ listed in Table 1. The Test portion consists of 497 tweets only, however, these were filtered down to 359 because the remaining have a *neutral* label which is not present in the training set.

Suicide and Depression Dataset The Suicide and Depression dataset [13] is collected from the Reddit platform, under different subreddits categories, namely “SuicideWatch”, “depression”, and “teenagers”.⁴ The texts of the posts from the “teenagers” category are labelled as negative, while the texts from the other two categories are labelled as positive. We excluded examples longer than 512 characters, then divided the dataset into three portions Train, Dev, and Test.

ChatGPT querying mechanism

We introduce the stages of querying ChatGPT as shown in Figure 1. The general mechanism to collect for our experiments is achieved by the following procedure for each problem:

- 1) Reformat all the texts of the Test portion of the dataset, by using a format that asks ChatGPT what is their guess about the label of the text.
- 2) Chunk the examples into 25 examples per chunk.
- 3) For each chunk, open a new ChatGPT Conversation.
- 4) Ask ChatGPT (manually) the reformatted question for each example, one-by-one, and collect the answers.
- 5) Repeat the steps 3–4 until the predictions for the whole Test set are finished.

²We acquired the dataset from <https://huggingface.co/datasets/sentiment140>, on 09.02.2023.

³<https://github.com/senticnet/chatgpt-affect>

⁴We acquired the dataset on 28.01.2023 from <https://www.kaggle.com/datasets/nikhileswarkomati/suicide-watch>

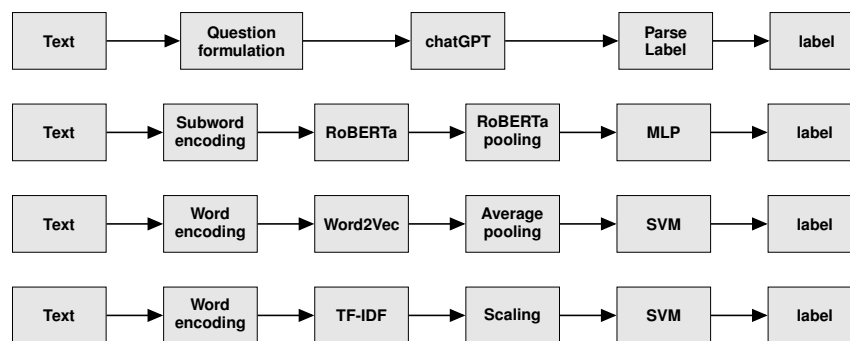


Figure 1. Pipelines of the ChatGPT (top), RoBERTa baseline (second), Word2Vec baseline (third), and BoW baseline (bottom) approaches.

6) Postprocess the results in case they need some cleanup.

The formatting in the first step and the post-processing in the last step are specified in the following two Subsections. We used the version of ChatGPT released on 30.01.2023⁵.

Question Formulation The formats that are used for the three problems are given by the following snippets. The example text is substituted in place of the *{text}* part, however, the quotation marks are kept since it specifies for ChatGPT that this a placeholder used by the question being asked. The formulations for the three problems are given by:

1) For the Big-five personality traits, we formulate the question:

“What is your guess for the big-five personality traits of someone who said “{text}”, answer low or high with bullet points for the five traits? It does not have to be fully correct. You do not need to explain the traits. Do not show any warning after.”

2) For the sentiment analysis, we formulate the question:

“What is your guess for the sentiment of the text “{text}”, answer positive, neutral, or negative? it does not have to be correct. Do not show any warning after.”

3) For the suicide problem, we formulate the following question:

“What is your guess if a person is saying “{text}” has a suicide tendency or not,

answer yes or no? it does not have to be correct. Do not show any warning after.”

The formulation of the question is of crucial importance to the answer ChatGPT will give; we encountered the following aspects:

- 1) Asking the question directly without asking about a guess made ChatGPT in many instances to answer that there is little information provided to answer the question, and it cannot answer it exactly. Hence, we ask it to guess the answer, and we declare that it is acceptable to be not fully accurate.
- 2) It is important to ask *what* the guess is and not *Can you guess*, because this can give a response similar to 1., where it responds with an answer that starts with *No, I cannot accurately answer whether...* Therefore, the question needs to be assertive and specific.
- 3) We need to specify the exact output format, because ChatGPT can get innovative otherwise about the formatting of the answer, which can make it hard to collect the answers for our experiment. Despite specifying the format, it still sometimes gave different formats. We elaborate in the next Subsection.
- 4) The questions for the suicide assessment task triggered warnings in the responses of ChatGPT due to its sensitive content. We elaborate on the terms of use in the Acknowledgement Section.

Parsing Responses The responses of ChatGPT need to be parsed, since ChatGPT can give arbitrary formats for a given answer, even when the content is the same. This is predominant in the personality traits, since there are five traits. Sometimes the answers are listed as bullet points,

⁵ChatGPT release notes <https://help.openai.com/en/articles/6825453-chatgpt-release-notes>

	RoBERTa			W2V	BoW
	N	U	α	C	η
O				0.0378	2.47×10^{-3}
C				0.0472	3.09×10^{-6}
E	2	498	5.66×10^{-4}	0.0069	1.09×10^{-5}
A				0.0218	4.65×10^{-4}
N				0.0657	2.21×10^{-6}
Sen	3	420	2.97×10^{-5}	0.0144	5.25×10^{-6}
Sui	3	497	8.04×10^{-4}	10.00	4.71×10^{-6}

Table 2. Hyperparameters of the different baselines. N is the number of hidden layers in the MLP using RoBERTa representations, U is the number of neurons in the first hidden layer (which is halved for each subsequent layers), and α is the learning rate. Adam optimiser always yields the best results as compared to SGD. C is the SVM parameter for Word2Vec, the sentiment model used linear kernel, while the other models used RBF kernel. η is the learning rate of the SGD in the BoW model.

other times they are all in one comma-separated line.

Also, it used different delimiters or order, e. g., “Openness: Low”, or “Low in Openness”, and “Low: Openness”. Additionally, in all problems, it sometimes gives an introduction for the answer, for example, “Here is my guess for ..”, or “Based on the statement”. We encounter this issue by using regular expressions to capture the responses.

Baselines

In order to compare the performance of ChatGPT on the different tasks, we need to use baselines and train them on the Train portion (while validating on the Dev portion). We employ three baselines, which serve as the specialised models specifically tailored for the corresponding downstream task. The first baseline is a robust language model (RoBERTa) trained on a large amount of text. The second is a simple baseline which uses a word model by employing pretrained Word2Vec embeddings on the words of a sentence, with a simple classifier. The third baseline is a simple Bag-of-Words (BoW) model that utilises a linear classifier. The hyperparameters of all models are optimised by selecting the hyperparameters yielding the best performance on the Dev portion. The hyperparameters are tuned using the SMAC toolkit [14], which is based on Bayesian Optimisation. The selected hyperparameters are listed in Table 2.

RoBERTa Language Model The baseline RoBERTa [15] is a pretrained BERT model, which has a transformer architecture. [15] trained two instances of RoBERTa; we use the smaller one, namely *RoBERTa-base*⁶, consisting of 110 million parameters. RoBERTa-base is pretrained on a mixture of several large datasets that included books, English Wikipedia, English news, Reddit posts, and stories [15]. The model starts by tokenising a text using subword encoding, which is a hybrid representation between character-based and word-based encodings. The tokens are then fed to RoBERTa to obtain a sequence of embeddings. The pooling layer of RoBERTa is then used to reduce the embeddings into one embedding only, hence acquiring a static feature vector of size 768 representing the text. We train additionally a Multi-Layer Perceptron (MLP) [16] to predict the final label. The pipeline for the model is shown in Figure 1.

For the training procedure, we use SMAC [14] to select the MLP specifications. We employ SMAC to sample a total of 100 models per task, and train them with a batch size 256 for 300 epochs with early stopping to prune the ineffective models. Eventually, the model with best performance on the Dev set is selected. The hyperparameter space consists of four hyperparameters: the number of hidden layers $N \in [0, 3]$, the number of neurons in the first hidden layer $U \in [64, 512]$ (log sampled), the optimisation algorithm (Adam [17] or Stochastic Gradient Descent (SGD) [16]), and the learning rate $\alpha \in [10^{-6}, 10]$ (log sampled). The number of neurons in the hidden layers is specified by the first one as a hyperparameter, then, the number of neurons is halved for each subsequent hidden layer (clipped to be at least 32). The hidden layers have Rectified Linear Unit (ReLU) as an activation function. The final layer has a sigmoid activation function. The loss function for classification is crossentropy, and mean absolute error for regression.

Word2Vec Word Embeddings The baseline Word2Vec [18], [19] makes use of pretrained word embeddings⁷, which are trained on a large

⁶Acquired on 09.02.2023 from https://huggingface.co/docs/transformers/model_doc/roberta

⁷Acquired on 16.02.2023 from <https://code.google.com/archive/p/word2vec/>

amounts of text from Google News. The model operates by tokenising a given text into words, each word is assigned an embedding from the pretrained embeddings. The embeddings are then averaged for all words to give a static feature vector of size 300 for the entire string. A Support Vector Machine (SVM) [16] is then used to predict the given task. The pipeline of this model is shown in Figure 1.

We train the SVM model by tuning its hyperparameter C using SMAC [14], by sampling 20 values within the range $[10^{-6}, 10^4]$ (log sampled), and choosing the model that yields the best score on the Dev set. We use Radial Basis Function (RBF) kernel for the SVMs, except for the sentiment dataset, where we apply a linear kernel as the sentiment dataset is much bigger (as shown in Table 1) which renders the RBF impractical due to the computational efficiency.

Bag of Words The Bag-of-Words (BoW) model is a very simple baseline, which does not rely on any knowledge transfer or large-scale training. In particular, it uses only in-domain data for training and no other data for either up- or downstreaming.

We utilise the classical technique term frequency – inverse document frequency (TF-IDF), which tokenises the sentences into words, then, a sentence is represented by a vector of the counts of the words it contains. The vector is then normalised by the term frequency across the entire Train set of the corresponding dataset. We restrict the words to the most common 10,000 words in the Train set, then we scale each feature to be within $[-1, 1]$, by dividing by the maximum absolute value of the feature across the Train set. We optimise a linear kernel SVM, we optimise using SGD [16] due to the high number of features (10,000 features). We tune the learning rate η of SGD using SMAC [14].

RESULTS

In this section, we review the results of our experiments. In summary, we evaluate the performance of ChatGPT against the three baselines RoBERTa, Word2Vec, and BoW on three downstream classification tasks, namely personality traits, sentiment analysis, and suicide tendency assessment. We measure classification accuracy

and Unweighted Average Recall (UAR) [20] as performance measures. UAR has an advantage of exposing if a model is performing very well on a class on the expense of the other class, especially in imbalanced datasets. Additionally, we utilise randomised permutation test as a statistical significance test [21]. The main results of the experiments are shown in Table 3.

Discussion

The RoBERTa model is achieving the best performance for the personality and suicide assessment tasks, with a statistically significant improvement of accuracy over ChatGPT. However, ChatGPT is the best in sentiment analysis, but only slightly better than RoBERTa. The UAR for the personality traits point to similar conclusions about the relative performance, however, they yield much lower values for all baselines on some of the traits (openness and agreeableness). The UAR measure generally yields similar results on all models for both sentiment analysis, and suicide assessment. The performance of ChatGPT on the personality assessment is inferior to the three baselines on the all traits. It is significantly worse than RoBERTa on all traits, and Word2Vec in three traits.

ChatGPT has the best performance in the sentiment analysis, where it is slightly better than RoBERTa and BoW and significantly better than Word2Vec. One of the potential reasons of the inferiority of Word2Vec and BoW on the sentiment dataset is not using subword encodings. The reason is that, the sentiment dataset is collected from Twitter, so it is very noisy, which can lead to many mistakes in identifying the words and hence assigning them the proper embeddings. Subword encoding avoids many of these issues, since few typos would still yield a meaningful subword representation of the given sentence.

The results on the suicide assessment problem show the contrast between the aforementioned analyses. The task is not as hard as the personality assessment problem, with a much bigger amount of training data. The suicide assessment can rather be thought of as classifying extreme negative sentiment, where [7] showed that ChatGPT is better at predicting negative sentiment than positive. However, the texts of the suicide dataset are much less noisy compared to the sentiment dataset. In that case, the performance of the

[%]	Accuracy				Unweighted Average Recall			
	ChatGPT	RoBERTa	Word2Vec	BoW	ChatGPT	RoBERTa	Word2Vec	BoW
O	46.6	66.0***	65.2***	59.7***	50.1	50.9	50.7	55.6
C	57.4	63.7*	62.7	55.6	57.7	60.8	60.0	56.3
E	55.2	66.0***	59.9	55.2	54.0	62.3***	55.5	53.7
A	44.8	67.4***	67.2***	58.5***	48.4	51.9	51.0	55.7*
N	47.2	62.1***	56.8***	56.0***	49.1	61.2***	54.6	55.8*
Sen	85.5	85.0	79.4*	82.5	85.5	85.0	79.4**	82.4
Sui	92.7	97.4***	92.1	92.7	91.2	97.4***	91.2	90.9

Table 3. The classification accuracy and Unweighted Average Recall (in %) of ChatGPT against the baselines on the different tasks (Sen: Sentiment, Sui: Suicide). The bold values show the best method for a combination of specific performance metric and prediction target. *, **, * indicate statistically significant difference as compared to ChatGPT, with p-values 0.05, 0.02, and 0.01, respectively. Significance tests are checked with a randomised permutation test.**

Word2Vec and BoW models are more or less on par with the ChatGPT model, while RoBERTa is significantly better than all of them.

Our experiments indicate that ChatGPT has a decent performance across many tasks (especially sentiment analysis, or similar), which is comparable to simple specialised models that solve a downstream task. However, it is not competent enough as compared to the best specialised model to solve the same downstream task (e.g., fine-tuned RoBERTa). The performance of ChatGPT does not generally show statistically significant differences when compared to the simplest baseline BoW, which does not make any use of pretraining. This is further confirmed by [8] in machine translation, and other tasks [7]. In summary, our study suggests that ChatGPT is a generalist model (in contrast to a specialised model), that can decently solve many different problems without specialised training. However, in order to achieve the best results on specific downstream tasks, dedicated training is still required. This might be enhanced in future versions of ChatGPT and alikes by including more diverse tasks for the Reinforcement Learning Human Feedback (RLHF) component in the training.

Limitations

The most crucial limitation of the presented results is the small amount of data for evaluation (497, 362, and 509 examples for the three tasks), since ChatGPT is only available for manual entries by the consumers and not for automated large-scale testing. Additionally, it only responds to approximately 25–35 requests per hour, in order to reduce the computational cost and avoid brute forcing. Another issue that may limit future experi-

ments is parsing the responses. In our experiments, ChatGPT responded with arbitrary formatting despite specifying the desired format explicitly in the question prompt.

CONCLUSION

In this paper, we provided first insight into the potential ‘full’ emergence of tasks by broad-data trained foundation models. We approached this from the perspective of natural language tasks in the Affective Computing domain, and chose ChatGPT as exemplary foundation model. To this end, we introduced a framework to evaluate the performance of ChatGPT as a generalist foundation model against specialised models on a total of seven classification tasks from three affective computing problems, namely, personality assessment, sentiment analysis, and suicide tendency assessment.

We compared the results against three baselines, which reflect training the downstream tasks, and using or not using additional data for the upstream task. The first model was RoBERTa, a large-scale-trained transformer-based language model, the second was Word2Vec, a deep learning model trained to reconstruct linguistic contexts of words, and the third was a simple bag-of-words (BoW) model.

The experiments have shown that ChatGPT is a generalist model that has a decent performance on a wide range of problems without specialised training. ChatGPT showed superior performance in sentiment analysis, and poor performance on personality assessment, and average performance on suicide assessment. In other words, we could demonstrate genuine emergence properties potentially rendering future efforts to collect task-specific databases increasingly obsolete.

However, the performance of ChatGPT is not particularly impressive, since it did not show statistically significant differences with a simple BoW model in almost all cases. On the other hand, RoBERTa fine-tuned for a specific task had significantly better performance as compared to ChatGPT on the given tasks, which suggests that despite the generalisation abilities of ChatGPT, specialised models are still the best option for optimal performance. However, this can be taken into consideration in future developments of foundation models like ChatGPT, in order to yield wider exploration spaces for training.

In the near future, we will extend our experiments to more metrics, e. g. , explainability and computational efficiency, on top of accuracy and UAR. We also plan to expand our comparative evaluation to more sophisticated models, e. g. , prompt-based classification [22] and neurosymbolic AI [23], more advanced affective computing tasks, e. g. , sarcasm detection [24] and metaphor understanding [25], but also more complex sentiment datasets requiring commonsense reasoning and/or narrative understanding.

ACKNOWLEDGEMENT

We would like to thank OpenAI for the usage of ChatGPT. We followed the policy of ChatGPT⁸. Our use of ChatGPT is purely for research purposes to assess emerging capabilities of foundation models, and does not promote the use of ChatGPT in any way that violates the aforementioned usage policy; in particular, with regards to the subject of self-harm; note that some of the examples in the datasets we use triggered a related warning by ChatGPT.

REFERENCES

1. R. Bommasani et al., "On the Opportunities and Risks of Foundation Models," *arXiv*, no. 2108.07258, 2021.
2. S. Mollman, "ChatGPT gained 1 million users in under a week. Here's why the AI chatbot is primed to disrupt search as we know it," 2022, <https://finance.yahoo.com/news/chatgpt-gained-1-million-followers-224523258.html>.
3. L. Ouyang et al., "Training language models to follow instructions with human feedback," *arxiv*, no. 2203.02155, 2022.
4. L. Gao, J. Schulman, and J. Hilton, "Scaling Laws for Reward Model Overoptimization," *arxiv*, no. 2210.10760, 2022.
5. A. Borji, "A Categorical Archive of ChatGPT Failures," *arxiv*, no. 2302.03494, 2023.
6. C. Zhou et al., "A Comprehensive Survey on Pretrained Foundation Models: A History from BERT to ChatGPT," *arxiv*, no. 2302.09419, 2023.
7. C. Qin et al., "Is ChatGPT a General-Purpose Natural Language Processing Task Solver?" *arxiv*, no. 2302.06476, 2023.
8. A. Hendy et al., "How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation," *arxiv*, no. 2302.09210, 2023.
9. V. Ponce-López et al., "Challearn lap 2016: First Round Challenge on First Impressions - Dataset and Results," *European conference on computer vision*, Springer, Springer International Publishing, Cham, Switzerland, 2016, pp. 400–418.
10. H. J. Escalante et al., "Design of an Explainable Machine Learning Challenge for Video Interviews," *International Joint Conference on Neural Networks (IJCNN)*, IEEE, Anchorage, AK, USA, 2017, pp. 3688–3695.
11. H. Kaya, F. Gurpinar, and A. Ali Salah, "Multi-Modal Score Fusion and Decision Trees for Explainable Automatic Job Candidate Screening From Video CVs," *Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, IEEE, Honolulu, HI, USA, 2017.
12. A. Go, R. Bhayani, and L. Huang, "Twitter Sentiment Classification using Distant Supervision," *CS224N project report*, Stanford, 2009, p. 2009.
13. V. Desu et al., "Suicide and Depression Detection in Social Media Forums," *Smart Intelligent Computing and Applications, Volume 2*, Springer Nature Singapore, Singapore, Singapore, 2022, pp. 263–270.
14. M. Lindauer et al., "SMAC3: A Versatile Bayesian Optimization Package for Hyperparameter Optimization," *Journal of Machine Learning Research*, 2022, pp. 1–9.
15. Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arxiv*, no. 1907.11692, 2019.
16. C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York City, NY, USA, 2006.
17. D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," *3rd International Conference on Learning Representations, Conference Track Proceedings*, ICLR, San Diego, CA, USA, 2015.
18. T. Mikolov et al., "Efficient Estimation of Word Representations in Vector Space," *arxiv*, no. 1301.3781, 2013.
19. T. Mikolov et al., "Distributed Representations of Words and Phrases and their Compositionality," *C. Burges*

⁸Usage policy released on 15.02.2023, <https://platform.openai.com/docs/usage-policies/disallowed-usage>

et al., eds., *Advances in Neural Information Processing Systems*, Curran Associates, Inc., Lake Tahoe, NV, USA, 2013.

20. B. Schuller et al., "The INTERSPEECH 2013 Computational Paralinguistics Challenge: Social Signals, Conflict, Emotion, Autism," *Proceedings INTERSPEECH*, ISCA, Lyon, France, 2013, pp. 148–152.
21. P. Good, *Permutation Tests: A Practical Guide to Resampling Methods for Testing Hypotheses*, Springer, New York City, NY, USA, 1994.
22. R. Mao et al., "The Biases of Pre-Trained Language Models: An Empirical Study on Prompt-based Sentiment Analysis and Emotion Detection," *IEEE Transactions on Affective Computing*, 2023.
23. E. Cambria et al., "SenticNet 7: A Commonsense-based Neurosymbolic AI Framework for Explainable Sentiment Analysis," *LREC*, 2022, pp. 3829–3839.
24. N. Majumder et al., "Sentiment and Sarcasm Classification with Multitask Learning," *IEEE Intelligent Systems*, , no. 3, 2019, pp. 38–43.
25. M. Ge, R. Mao, and E. Cambria, "Explainable Metaphor Identification Inspired by Conceptual Metaphor Theory," *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, 2022, pp. 10681–10689.

Mostafa M. Amin is currently working toward the Ph.D. degree with the Chair of Embedded Intelligence for Health Care and Wellbeing with University of Augsburg, while working as Senior Research Data Scientist at SyncPilot GmbH in Augsburg, Germany. His/her research interests include Affective Computing, Audio and Text Analytics. Amin received a MS.c. degree in Computer Science from the University of Freiburg, Germany. Contact him at mostafa.mohamed@uni-a.de

Erik Cambria is an associate professor at Nanyang Technological University, Singapore. His research focuses on neurosymbolic AI for explainable natural language processing in domains like sentiment analysis, dialogue systems, and financial forecasting. He is an IEEE Fellow and a recipient of several awards, e.g., IEEE Outstanding Career Award, was listed among the AI's 10 to Watch, and was featured in Forbes as one of the 5 People Building Our AI Future. Contact him at cambria@ntu.edu.sg.

Björn W. Schuller is currently a professor of Artificial Intelligence with the Department of Computing, Imperial College London, UK, where he heads the Group on Language, Audio and Music (GLAM). He is also a full professor and the head of the Chair of

Embedded Intelligence for Health Care and Wellbeing with the University of Augsburg, Germany, and the Founding CEO/CSO of audEERING. Contact him at schuller@IEEE.org.