Department: Affective Computing and Sentiment Analysis
Editor: Erik Cambria, Nanyang Technological University

Can ChatGPT's Responses Boost Traditional Natural Language Processing?

Mostafa M. Amin
University of Augsburg; SyncPilot GmbH

Erik Cambria
Nanyang Technological University

Björn W. Schuller
University of Augsburg; Imperial College London

***Abstract*—The employment of foundation models is steadily expanding, especially with the launch of ChatGPT and the release of other foundation models. These models have shown the potential of emerging capabilities to solve problems, without being particularly trained to solve. A previous work demonstrated these emerging capabilities in affective computing tasks; the performance quality was similar to traditional Natural Language Processing (NLP) techniques, but falling short of specialised trained models, like fine-tuning of the RoBERTa language model. In this work, we extend this by exploring if ChatGPT has novel knowledge that would enhance existing specialised models when they are fused together. We achieve this by investigating the utility of verbose responses from ChatGPT about solving a downstream task, in addition to studying the utility of fusing that with existing NLP methods. The study is conducted on three affective computing problems, namely sentiment analysis, suicide tendency detection, and big-five personality assessment. The results conclude that ChatGPT has indeed novel knowledge that can improve existing NLP techniques by way of fusion, be it early or late fusion.**

WITH THE RECENT rapid growth of foundation modelsas large language models (LLMs), a potential has appeared for emerging capabilities [1] of such models to perform new downstream tasks or solve new problems, that they were not particularly trained on in the first place. This includes models like GPT-3.5 [2], and RoBERTa [3].

The capabilities of such foundation models are being explored in various domains, like affective computing [4], and sentiment analysis [5]. The phenomenon of emerging capabilities of LLMs [1] was more pronounced with the utilisation of fine-tuning techniques like Reinforcement Learning with Human Feedback (RLHF), as it was

employed in InstructGPT [2], which was later included in GPT-3.5 and GPT-4 models, the main underlying models of ChatGPT. In a previous study [4], we studied the emerging capabilities of ChatGPT to solve affective computing problems, as compared to *specialised* models trained on a particular problem. The study has indeed shown the emergence of such capabilities in affective computing problems. like sentiment analysis, suicide tendency detection, and personality traits assessment. The performance was comparable to classical Natural Language Processing (NLP) models like Bag-of-Words (BoW) [6], but not better than fine-tuned LLMs like RoBERTa [3]. Another issue that was encountered was parsing the results from the responses of ChatGPT, since it frequently formatted the responses differently despite being prompted to respond with a specific format. The aforementioned conclusions had a follow up question, whether foundation models contain novel knowledge that is not acquired by specialised training of NLP models, hence leading to better results in the scenarios when fusing foundation models with specialised models. We mainly investigate this question in this study. The contributions of this paper are as follows:

1) We introduce how to prompt ChatGPT to give verbose responses that solve affective computing problems, we demonstrate this in sentiment analysis, suicide and depression detection, and big-five personality traits assessment.
2) We present the utility of employing the verbose responses of ChatGPT when they are processed with traditional NLP techniques.
3) We introduce how to fuse ChatGPT with existing NLP methods for affective computing, and investigate their different combinations with different fusion methods.

The remainder of the paper is organised as follows: in the next section, we discuss related work; then, we introduce our method; afterwards, we present and discuss the results; finally, we propose concluding remarks.

RELATED WORK

We focus on related work within the area of foundation models in affective-computing-related tasks (in the text domain) or hybrid formulations between foundation models and traditional NLP methods. Both [7], [8] explore a fusion between ChatGPT and other transformer-based models for Named Entity Recognition (NER). [9] investigates the capabilities of ChatGPT on various NLP tasks including affective computing tasks. [5] investigates the performance of ChatGPT in several in sentiment analysis and aspect extraction.

Dataset		Train	Dev	Test	+ve	-ve
Sentiment		20,000	5,000	3,000	1,516	1,484
Suicide		9,999	3,881	2,375	757	1,618
Personality	O	5,992	2,000	1,997	1,336	661
	C				1,133	864
	E				890	1,107
	A				1,332	665
	N				1,122	875

Table 1: Datasets statistics, including counts of positive and negative classes in the Test set.

METHOD

In this section, we present first the datasets for the different affective computing problems. Afterwards, we introduce the prompting of ChatGPT, then the methods for extracting features. Subsequently, we present how we train and tune the machine learning models. Finally, we present a simple baseline based on ChatGPT responses. The pipeline of our method is presented in Figure 1.

Datasets

We present here the adopted datasets for the three affective computing problems. A summary of their statistics is in Table 1.

Sentiment Dataset We make use of the Twitter Sentiment140 dataset [10] for sentiment analysis.[1] The dataset consists of tweets that were collected from Twitter. Tweets are generally very noisy texts. The dataset consists of tweets and the corresponding binary sentiment labels (positive, or negative). The original dataset consists of 1,600,000 Tweets, however, we filtered these down into a total of 28,000 examples. [2] We do not make use of the original Test portion in the dataset, since it consists of only 497 Tweets, and it also contains a 'neutral' label unlike the rest of the dataset. We split the original training portion into three parts as shown in Table 1.

[1]We acquired the dataset from https://huggingface.co/datasets/sentiment140, on 09.02.2023.

[2]https://github.com/mostafa-mahmoud/chat-gpt-fusion-evaluation

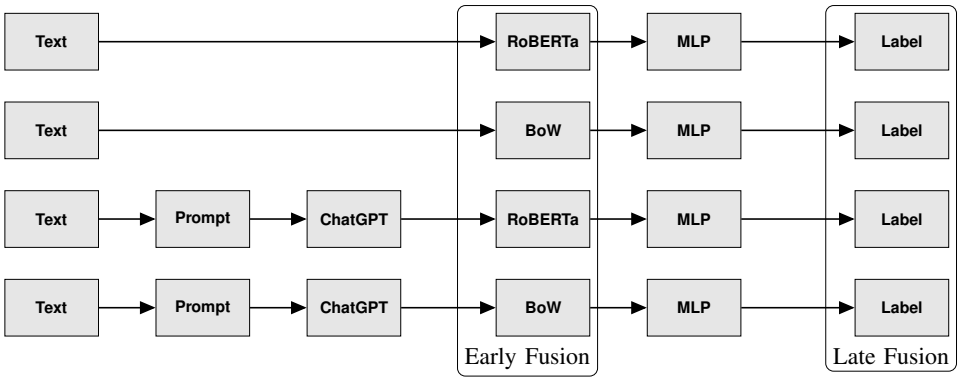

Figure 1: Pipelines of the different fusion methods. Each branch shows a single modality of selecting an input text and processing it with an NLP technique. The input text is either used directly or by using a corresponding response from ChatGPT about it. Subsequently, it is processed by RoBERTa or BoW. MLPs are then used on the features to predict the binary classification labels. We select specific branches to carry out different fusion methods. For early fusion, the features from the selected branches are concatenated, then one MLP is used on that to predict a label. For late fusion, the prediction scores from the single branches are averaged to give a classification probability.

Suicide and Depression Dataset The Suicide and Depression dataset [11] was gathered from the platform Reddit. The collection was gathered under different categories (subreddits), namely "depression", "SuicideWatch", and "teenagers".[3] The 'non-sucide' label was given to the posts from the "teenagers" category, while the remaining texts were given the label 'suicide'. After excluding examples longer than 512 characters and downsampling the dataset, we acquired a dataset of size 16,266 that we divide into three portions Train, Dev, and Test as shown in Table 1, since the original dataset was not split.

Personality Dataset We make use of the First Impressions (FI) dataset [12]. for the personality task[4]. The big-five personality traits (OCEAN) are the traits used to represent personality, namely, *Openness to experience*, *Conscientiousness*, *Extraversion*, *Agreeableness*, and *Neuroticism*. The dataset was gathered by collecting videos from YouTube, and slicing them into 15 seconds clips with one speaker. In our setup, we utilise only the text modality of the entire FI dataset, with its provided split, originating from the transcriptions of the videos. Each personality trait is represented by a continuous regression value within $[0, 1]$. We train regression models (by employing Mean Absolute Error as a loss function), because the continuous labels give a granular estimation of the personality labels. For evaluation, we binarise the labels by the threshold 0.5.

[3]We acquired the dataset on 28.01.2023 from https://www.kaggle.com/datasets/nikhileswarkomati/suicide-watch

[4]We acquired the dataset on 03.02.2023 from https://chalearnlap.cvc.uab.cat/dataset/24/description/

ChatGPT Prompts

To formulate the ChatGPT text modalities, we need to formulate a prompt to ask ChatGPT, in order to obtain a reasonable answer. We formulate a prompt for each specific problem to ask it about the label. First, we design the prompt to ask for a binary label of the corresponding problem, while emphasising narrowing down the answer to only two labels while excluding more 'neutral' labels. Similar to a previous work [4], we design the prompts to have the disclaimer *It does not have to be fully correct*, and ask *what is your guess for the answer*, instead of *What is the answer* or *Can you guess the answer*. This formulation is to avoid ChatGPT from responding that it is not sure about the answer, hence not giving any answer. Unlike [4], we ask ChatGPT to be verbose and explain the reasoning behind the answer, since we are processing that with NLP methods (unlike [4], where the final label was parsed). A last sentence is added to avoid redundant disclaimer in the response of ChatGPT.

We make use of the OpenAI API to use ChatGPT[5], using the model 'gpt-3.5-turbo-0301'. We do not give a system message, we just use the prompt corresponding to the specific problem as the only user message in the input conversation, with the input text of the example. The assistant response is what we use as the response of ChatGPT. We use the default parameters for generation, namely the answer with highest score ($n = 1$), and the temperature parameter $T = 1.0$.

The prompts for the given problems are given below, by substituting the input $\{text\}$. For the personality traits, we query the API five times for each of the five traits by substituting the $\{trait\}$.

- The prompt for the sentiment classification:

  What is your guess for the sentiment of the text "{text}"? Answer positive or negative, but not neutral. Try to narrow down the answer to be one of those two. It does not have to be fully correct. Explain your answer briefly. Do not show any warning after.

- The prompt for the suicide detection:

  What is your guess, is a person saying the text "{text}" has suicide tendencies? Answer yes or no. It does not have to be fully correct. Explain your answer briefly. Do not show any warning after.

- The prompt for the personality traits:

  What is your guess for the personality trait "{trait}", from the big-five personality traits, of someone who said "{text}"? Answer low or high, but not neutral. Try to narrow down the answer to low or high. It does not have to be fully correct. Explain your answer briefly. Do not show any warning after.

Text Features

RoBERTa Language Model The RoBERTa [3] feature set is obtained by the pretrained LLM RoBERTa, which is based on the BERT model with a transformer architecture. The model has two variants; we utilise the smaller variant, namely RoBERTa-base[6]. The model was trained on large datasets with reddit posts and English Wikipedia, and English news [3]. In order to extract the embedding for a string, it is first encoded with a subword encoder then fed to the RoBERTa model to give a sequential set of features with attention weights. These are reduced through a pooling layer in the model to produce the final static vector of 768 features representing the given string.

Bag-of-Words The BoW feature set is achieved by constructing n-grams, and then using the classical term-frequency inverse-document-frequency (TF-IDF) to count each term while normalising them by the frequency across all documents [6]. For the input texts, we keep only the most common 10,000 words (i. e. , 1-grams), to give a static vector of 10,000 features representing the text. For the responses of ChatGPT, we utilise the most common 2,000 n-grams ($n \in \{1, 2, 3\}$). The vectors are scaled by the maximum absolute values to be within the range $[-1, 1]$. The reason we utilise n-grams for ChatGPT responses is that, the responses usually include expressions like 'high extraversion', or 'sentiment is negative'.

Models and tuning

Given a feature set (or a fusion thereof) we train a Multi-Layer Perceptron (MLP) [6] to predict the final label. We construct an MLP with N hidden layers, with U units in the first hidden layer, then each following hidden layer has half the number of neurons of the hidden layer preceding it (we cap this number to be at least 32 units). ReLU is the activation function used for all layers, except for the final layer, where we apply sigmoid to predict the final label within the range $[0, 1]$. We leverage the Adam optimisation algorithm with a learning rate α. The loss function is either Mean Absolute Error (MAE) for regression training (for personality training), or otherwise negative log likelihood for classification training. We employ the hyperparameter optimisation toolkit SMAC [13] to select the best hyperparameters for each problem/dataset and each input modality (or early fusion combinations thereof). We explore 20 hyperparameters samples for each problem. The hyperparameter space has $N \in [0, 3]$, $U \in [64, 512]$ (log-sampled), and $\alpha \in [10^{-6}, 10]$ (log-sampled).

[5]https://platform.openai.com/docs/guides/gpt/chat-completions-api

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

Fusion

We deploy early fusion by concatenating the features extracted by RoBERTa or BoW, then training one MLP on the concatenated vector similar to training a single method. On the other hand, the late fusion is achieved by averaging the probabilities predicted by the given methods.

Baseline

We employ a simple baseline based on the responses of ChatGPT, which we prompt ChatGPT to give a binary label before explaining the answer, hence we construct the baseline to predict a label only if the word corresponding to its class is present in the response. For sentiment analysis, the baseline would predict 'positive' only if the response contains the word 'positive', and it would predict 'negative' only if the response contains the word 'negative'. For suicide detection, the two classification keywords are 'yes' and 'no'. For personality, the two keywords become 'high' and 'low'. We exclude the evaluation of responses that include both words or neither, which is roughly only 5% of the Test sets in our experiments. The intuition behind this baseline is that it is similar to parsing the labels from the non-verbose response.

RESULTS

We experiment the combinations of three main parameters, the *text* to be used, the corresponding extracted *features* to represent the text, and *how to fuse* them. The main results of the experiments are shown in Table 2. Finally, we refer to the combination of input text (original input or ChatGPT response thereof) and NLP processing technique as a *modality*.

Discussion

The results of utilisng the original text (for each of the single modalities Text+RoBERTa and Text+BoW) are close to previous work [4], with a slight difference due to the different sampling from the original datasets. The results of the single modality ChatGPT+RoBERTa are decent, comparable to the single modality Text+BoW, but worse than Text+RoBERTa in most cases except for sentiment analysis. The results of ChatGPT+BoW are slightly worse than ChatGPT+RoBERTa. In a similar fashion, these results of ChatGPT are resembling the previous work [4], where ChatGPT was comparable to the Text+BoW modality. Furthermore, the aggregate performances across problems is also similar to [4], where ChatGPT was the most superior in sentiment analysis, whilst most inferior in personality assessment.

The results of fusion are inclined to show that the most competent fusion combination is adopting only Text+RoBERTa and ChatGPT+RoBERTa, whether in early or late fusion; however, the early fusion of these two modalities is showing the most superior performance in most scenarios, except the sentiment analysis. Disregarding the specific combination of these two modalities, late fusion is performing better compared to the corresponding instances of early fusion in most cases of the other modality combinations. For instance, the late fusion of all modalities is better than their early fusion; similarly for the combination of Text+RoBERTa and Text+BoW.

Consequently, the impact of fusion overall is not very straight forward to explain, because the *single* modality Text+RoBERTa is the best for the personality assessment, while the *early* fusion of Text+RoBERTa and ChatGPT+RoBERTa is the best for suicide detection, and the *late* fusion of all modalities is the best for sentiment analysis. The reason for the superiority of the single modality in the personality assessment is probably due to the poor performance of ChatGPT on the given text, since ChatGPT single modalities are the worst ones. On the other hand, if ChatGPT has a decent performance, then applying fusion has definitely a strong improvement impact, be it early or late fusion. However, the superiority of late fusion against early fusion depends primarily on the problem and the data distribution. From the practical advantages of early fusion, it needs hyperparameter tuning only once, compared to the late fusion which needs to tune a model for each modality. On the other hand, the late fusion has an architectural advantage that it can deploy different training sizes for each modality.

In our previous work [4], ChatGPT results were labels that were parsed from the non-verbose responses (typically, a binary label like 'low' or 'high', with some variance in the formatting), whereas in this work we process the verbose response by applying NLP methods. The effectiveness of employing the verbose responses is demonstrated by the baseline approach, where the

Text		ChatGPT		Fusion	Sent.	Suic.	Personality					
RoBERTa	BoW	RoBERTa	BoW				Average	O	C	E	A	N
Baseline				–	77.68	94.48	54.34	65.54	58.43	47.89	53.35	46.47
✓				–	77.83	**95.37**	**64.12**	**67.55**	**63.09**	**61.19**	**67.55**	**61.19**
	✓			–	73.90	90.40	61.66	66.80	59.89	57.34	66.80	57.49
		✓		–	**80.27**	92.34	61.01	66.90	59.09	55.43	66.70	56.94
			✓	–	79.83	91.92	60.71	66.90	57.24	55.73	66.70	56.99
✓		✓		Early	**81.20**	**96.17**	**63.65**	**68.15**	**61.84**	**60.54**	66.70	**60.99**
	✓		✓	Early	80.90	93.52	61.79	66.90	60.39	56.94	66.60	58.14
✓	✓			Early	76.27	92.97	62.21	67.40	59.69	59.39	66.60	57.99
		✓	✓	Early	80.03	91.96	60.89	66.90	58.29	55.53	66.60	57.14
✓	✓	✓	✓	Early	80.93	93.94	61.53	67.00	60.34	57.04	**66.80**	56.48
✓		✓		Late	81.60	**96.13**	63.26	**66.95**	61.19	59.49	66.70	**61.99**
	✓		✓	Late	80.77	93.94	61.68	66.90	59.94	58.54	66.65	56.38
✓	✓			Late	79.40	95.54	**63.59**	66.75	**63.40**	**60.79**	**66.75**	60.24
		✓	✓	Late	81.13	92.76	61.08	66.90	59.64	55.38	66.65	56.84
✓	✓	✓	✓	Late	**82.60**	95.45	62.66	66.90	61.49	59.39	66.70	58.84

Table 2: Classification accuracy results for all the problems with the different fusion methods. There are two text-based inputs, the original text (Text), or the verbose response of ChatGPT on a question about the original text and the corresponding problem (ChatGPT). Each text input is processed in two ways, using RoBERTa features or BoW. The features are processed with an MLP to give the final binary classification label of the problem. The fusion is either done on the feature level with one MLP (Early), or on the predictions level (Late). Marked in bold are the best results for each combination of problem and fusion. Underlined are the best results for each problem.

results of the single ChatGPT modalities are close to the baseline. The verbose responses (compared to the non-verbose ChatGPT baseline) lead to better responses for both sentiment analysis and personality assessment, but with some drop in suicide detection. The verbose responses have the additional advantage of avoiding the problem of parsing the label from the response of ChatGPT, since the responses (including the non-verbose) do not always follow the same format despite being prompted to [4]. The last obvious advantage of verbose responses is the ability to include them in fusion models in various ways, which can lead to a much better performance as discussed earlier.

CONCLUSION

In this work, we explored the fusion capabilities of ChatGPT with traditional Natural Language Processing (NLP) models in affective computing problems. We first prompted ChatGPT to give verbose responses to answer binary classification questions for three affective computing downstream tasks, namely sentiment analysis, suicide tendency detection, and big-five personality traits assessment. Additionally, we processed the input texts and the corresponding ChatGPT responses with two NLP techniques, namely fine-tuning RoBERTa language model and n-gram BoW; these features were trained by leveraging Multi-Layer Perceptrons (MLPs). Furthermore, we investigated two fusion methods, early fusion (on the features level) or late fusion (on the prediction level).

The experiments have demonstrated that leveraging ChatGPT verbose responses bears novel knowledge in affective computing and probably beyond, which should be evaluated next, that can aid existing NLP techniques by ways of fusion, whether early or late fusion. First, we demonstrated the benefit of using verbose responses while processing them with NLP techniques, as compared to parsing classification labels from the non-verbose labels. Subsequently, this provided the possibility of seamlessly fusing ChatGPT responses with existing NLP methods, hence achieving a better performance via both early or late fusions. Furthermore, the experiments have demonstrated that utilising only RoBERTa to process and fuse the input texts and ChatGPT responses (with an inclination to early fusion than late) can be sufficient to reach the best performance.

REFERENCES

1. J. Wei et al., "Emergent Abilities of Large Language Models," *arXiv:2206.07682*, 2022.
2. L. Ouyang et al., "Training language models to follow instructions with human feedback," *arXiv:2203.02155*, 2022.

3. Y. Liu et al., “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” *arXiv:1907.11692*, 2019.
4. M. M. Amin, E. Cambria, and B. W. Schuller, “Will Affective Computing Emerge from Foundation Models and General AI? A First Evaluation on ChatGPT,” *IEEE Intelligent Systems*, vol. 38, no. 2, 2023, pp. 15–23.
5. W. Zhang et al., “Sentiment Analysis in the Era of Large Language Models: A Reality Check,” *arXiv:2305.15005*, 2023.
6. C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, New York City, NY, USA, 2006.
7. Y. Chen, V. Shah, and A. Ritter, “Better Low-Resource Entity Recognition Through Translation and Annotation Fusion,” *arXiv:2305.13582*, 2023.
8. J. Li et al., “Prompt ChatGPT In MNER: Improved multimodal named entity recognition method based on auxiliary refining knowledge from ChatGPT,” *arXiv:2305.12212*, 2023.
9. J. Kocoń et al., “ChatGPT: Jack of all trades, master of none,” *Information Fusion*, vol. 99, 2023, p. 101861.
10. A. Go, R. Bhayani, and L. Huang, “Twitter Sentiment Classification using Distant Supervision,” *CS224N project report, Stanford*, 2009, p. 2009.
11. 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.
12. V. Ponce-López et al., “Chalearn lap 2016: First Round Challenge on First Impressions - Dataset and Results,” *European conference on computer vision*, Springer International Publishing, Cham, Switzerland, 2016, pp. 400–418.
13. M. Lindauer et al., “SMAC3: A Versatile Bayesian Optimization Package for Hyperparameter Optimization,” *Journal of Machine Learning Research*, vol. 23, no. 54, 2022, pp. 1–9.

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 research interests include Affective Computing, Audio and Text Analytics. He received a M.Sc. degree in Computer Science from the University of Freiburg, Germany. Contact him at mostafa.mohamed@uni-a.de

Erik Cambria is a professor of Computer Science and Engineering at Nanyang Technological University, Singapore. His research focuses on neurosymbolic AI for explainable sentiment analysis in domains like social media monitoring, financial forecasting, and AI for social good. 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, & 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. He is an IEEE Fellow alongside other Fellowships. Contact him at schuller@IEEE.org.