Generating FER models using ChatGPT

Radu-Daniel BOLCAȘ

Faculty of Electronics, Telecommunications and Information Technology, National University of Science and Technology Politehnica Bucharest, Romania

radu.bolcas@gmail.com

Abstract: This paper investigates the utilisation of ChatGPT (Generative Pre-trained Transformer) in the field of Facial Emotion Recognition (FER), streamlining development processes and making them smoother and faster. With ChatGPT, coding and debugging happen fast, leading to the swift creation of performing models in mere minutes. It highlights the transformative potential of integrating ChatGPT into image processing workflows, accelerating progress in computer vision, graphics, and multimedia. This paper also shows the importance of word choice and developer skill in finding the right balance between speed and accuracy. It sets the stage for further investigations into advanced Large Language Models (LLMs) and fine-tuning parameters, leading towards faster, more efficient image processing techniques.

Keywords: Facial Emotion Recognition (FER), Deep Neural Networks, Convolutional Neural Networks (CNN), Computer Vision, Machine Learning, Large Language Model (LLM), ChatGPT(Generative Pre-trained Transformer).

Generarea de modele FER utilizând ChatGPT

Rezumat: Această lucrare investighează utilizarea ChatGPT (Generative Pre-trained Transformer) în domeniul recunoașterii emoțiilor faciale, eficientizând procesele de dezvoltare, făcându-le mai ușoare și mai rapide. Cu ChatGPT, scrierea de cod și depanarea se realizează rapid, ceea ce duce la crearea rapidă a unor modele performante în doar câteva minute. Acest proces evidențiază potențialul integrării ChatGPT în procesele de prelucrare a imaginilor, accelerând progresul în domeniile vederii computerizate, graficii și multimedia. Această lucrare arată, de asemenea, importanța alegerii cuvintelor și a abilităților dezvoltatorului în găsirea echilibrului corect între viteză și precizie. Acest aspect pregătește terenul pentru cercetările ulterioare în ceea ce privește modelele de limbaj mari (LLM) și reglarea fină a parametrilor, ceea ce duce la tehnici de procesare a imaginilor mai rapide și mai eficiente.

Cuvinte cheie: recunoașterea emoțiilor faciale, rețele neuronale adânci, rețele neuronale convoluționale, vedere computerizată, învățare automată, modele de limbaj mari, ChatGPT.

1. Introduction

Facial Expression Recognition (FER) is a computer vision endeavour designed to discern and classify emotional expressions conveyed through human facial features. Its objective is to determine in real-time emotions by analysing facial components such as eyebrows, eyes, and mouth, thereby associating them with a predefined set of emotions comprised of anger, fear, surprise, sadness, and happiness.

The ability to recognise emotions from images is a fundamental aspect of various applications ranging from automated diagnosis and mental health to improving customer experiences or security implementations.

For decades, psychologists have shown keen interest in detecting human emotions. Among the studies done, Ekman and Friesen determined a number of basic emotions that are expressed consistently across cultures and societies (Ekman & Friesen, 1971). These emotions are anger, joy, disgust, surprise, sorrow, and fear.

The FER machine learning research community primarily operates within two paradigms: supervised and unsupervised learning. While supervised learning has been the dominant approach in FER research, there has been limited exploration of its counterpart, unsupervised learning.

Unsupervised learning in facial emotion recognition refers to a subset of machine learning techniques where algorithms are trained on input data without explicit supervision or labelled output. In this context, unsupervised learning methods aim to identify patterns or structures within the data itself without the need for predefined labels or categories.

https://doi.org/10.33436/v34i2y202407

Convolutional Neural Networks (CNNs) have been extensively employed in the field of Facial Expression Recognition (FER) as the supervised learning method and have yielded favourable results. While the performance is excellent, the training process proves to be time-consuming, requiring a substantial dataset and numerous layers and neurons to effectively extract information from the dataset.

The development of CNN models involves a series of iterative stages for achieving accurate and efficient results in various applications. The first step in building a CNN model involves acquiring a suitable dataset. Depending on the specific task, researchers may choose to utilise existing datasets or create their own by collecting and annotating images relevant to the problem at hand.

Once the dataset is obtained, preprocessing steps are essential to ensure that the data is standardised, cleaned, and ready for training. This may involve tasks such as resizing images to a uniform size, normalising pixel values, and augmenting the dataset with variations to enhance the model's robustness and generalisation capabilities.

The choice of model architecture plays a crucial role in determining the performance of the CNN model. Researchers can opt to use established architectures such as VGG (Simonyan & Zisserman, 2014), ResNet (He et al., 2015) or others as a starting point, or they may design custom architectures tailored to the specific requirements of the task. The selected architecture should be a balance between complexity and efficiency, considering factors such as computational resources and the size of the dataset.

With the dataset prepared and the model architecture defined, the next step is to train the CNN model to learn to identify relevant patterns and features from the input data.

Hyperparameters are configuration settings that govern the training process and impact the performance of the model. Tuning these hyperparameters, such as learning rate, batch size, and regularisation techniques, is essential for optimising the model's performance and preventing overfitting or underfitting.

Once the model is trained, it is evaluated on a separate validation or test dataset to assess its performance. Metrics such as accuracy, precision, recall, and F1-score are commonly used to measure the model's effectiveness in correctly classifying images. Evaluation results provide insights into the model's strengths and weaknesses and guide further refinements.

The proposed approach involves leveraging OpenAI's ChatGPT (OpenAI, 2024) to generate the initial portion of the code, thereby allowing for more focused attention on subsequent stages. By employing ChatGPT as a supportive tool, the duration required for development and initial validation is significantly reduced. Upon completing all the steps and obtaining the results of the generated values, it can be assessed whether the model serves as a suitable starting point or if adjustments to the architecture are necessary.

The structure of this paper is as follows: Section 2 provides an overview of the FER and LLM background, Section 3 delves into the implementation and the results, Section 4 analyses the findings, and Section 5 outlines the conclusions.

2. Background and related work

2.1. Large Language Models (LLMs)

Large Language Models (LLM) are artificial intelligence models that are trained on vast amounts of text data to understand and generate human-like language. These models, such as ChatGPT (Generative Pre-trained Transformer) models developed by OpenAI, are capable of processing and generating text in a way that closely resembles human language patterns. LLMs have the ability to understand context, semantics, and syntax in text, enabling them to perform a wide range of natural languages processing tasks, such as language translation, text summarisation, question answering, and more. They achieve this through a complex network architecture that learns from patterns in the data during training.

LLMs undergo a comprehensive training process utilising vast amounts of text data sourced from diverse mediums such as books, articles, and websites. This corpus serves as the foundation for teaching the model the intricate statistical patterns and structures inherent in natural language. Prior to processing, text is segmented into smaller units known as tokens, which can range from individual words to characters depending on the model's configuration. These tokens are then transformed into high-dimensional vector representations called embeddings, capturing semantic and syntactic nuances essential for understanding context. LLMs are constructed atop the Transformer architecture, comprising multiple layers of self-attention mechanisms enabling the model to discern the significance of various words within a sentence. Through a defined training objective, LLMs learn to predict token likelihoods based on the surrounding context, often employing techniques like masked language modelling. Subsequently, fine-tuning allows LLMs to specialise in specific tasks or domains by adjusting parameters and incorporating task-specific examples. Once trained, LLMs can generate text by iteratively predicting subsequent tokens within a sequence, guided by preceding context until reaching predefined termination criteria. Evaluation of LLMs centres on their ability to produce coherent and contextually relevant text, assessed through tasks such as language translation, sentiment analysis, and text summarisation.

These models have gained significant attention and popularity with researchers due to their impressive capabilities in generating coherent and contextually relevant text.

In a study conducted by Wake et al., they employed ChatGPT to assess its proficiency in detecting emotions conveyed through text (Wake et al., 2023). Utilising datasets such as IMOCAP (Busso et al., 2008), MELD (Poria et al., 2018), EmoryNLP (Zahiri & Choi, 2018) and DailyDialog (Li et al., 2017), they obtained results surpassing chance levels, consistent with prior research. They investigated two scenarios: one where they utilised the prompt output directly and another where they conducted fine-tuning. Discrepancies were observed across different emotion labels, with variations even within the same label across different datasets. Fine-tuning proved effective in enhancing accuracy. Furthermore, they noted sensitivity to changes in label names, such as “happiness” and “happy”. Notably, the EmoryNLP dataset achieved an F1 score of 0.35, while MELD reached 0.68, showcasing ChatGPT's potential beyond conventional information retrieval tasks.

In another study a different approach was used by Blocklove et al., where they paired an engineer with ChatGPT-4 as an aide to create a complex hardware design with few conversations, although they employed the replay function in some cases (Blocklove et al., 2023). Given the limit of messages for ChatGPT-4, the total time for generating was 22.8 hours while also including the restarts. If the limit is disregarded, it’s estimated in the study that the needed time would be less than 100 minutes.

The efficacy of integrating Large Language Models (LLMs) into programming practices is further demonstrated in an article detailing Google's collaboration with over 10,000 software developers, who utilised Google's proprietary LLM (Tabachnyk & Nikolov, 2022). The study revealed significant improvements in developer productivity, evidenced by a reduction in coding iteration duration (time between builds and tests).

Ahmad et al. conducted a study to explore ChatGPT's capability to collaborate with human developers in designing software architecture (Ahmad et al., 2023). They observed variability in responses, suggesting different architectures, which could be mitigated through iterative dialogue with ChatGPT to refine architecture requirements and outputs. Another notable finding was that ChatGPT lacks sensitivity to ethics and intellectual property concerns, potentially generating code that collects user data or produces non-compliant software, necessitating developer oversight. Additionally, biases may arise from widespread solutions or discrepancies in training data classes, leading to suggestions of suboptimal architectures. The study highlighted various types of validity to consider: internal validity regarding the systematic interaction with ChatGPT via its prompt, external validity to assess the generalizability of findings to different contexts, and conclusion validity regarding the suitability of the study approach.

Murr et al. conducted a study exploring various approaches to employing prompts for LLMs (Murr, Grainger & Gao, 2023). Utilising test-driven development (TDD), the authors examined several prompt styles. "Prompt Only" entails presenting the problem with minimal information.

"Prompt with Tests" includes the problem statement along with example test cases. "Prompt Tests Only" provides only the test cases with their respective names. "Prompt Generic Tests" offers test cases with a masked function name. The most efficient approach was "Prompt Tests Only", with 231 tests passed out of a total of 315. Following closely was "Prompt with Tests", with 228 tests passed out of 315. "Prompt Only" demonstrated a slightly lower performance, with 222 tests passed out of 315, while the least effective method was "Prompt Generic Tests", with 200 tests passed out of 315.

2.2. FER architecture and databases

The database used for this study was FER2013 which is a dataset extensively used in the research community (Goodfellow et al., 2013). The FER2013 dataset contains 35,888 images depicting seven distinct emotions: anger, neutral, disgust, fear, happiness, sadness, and surprise. These images are categorised into three subsets: training, validation, and testing.

One of the aspects of any machine learning endeavour is selecting a model and a database to use. Models can vary in accuracy and multiple layers are not always the solution. In some cases a hyper-parameterized model can outperform more complex ones. This is the case for Khaireddin and Chen, who utilised a convolutional neural network model and adjusted its hyperparameters (Khaireddin & Chen, 2021). The network architecture comprises four convolutional layers and three fully connected layers. Within each convolutional layer, there are two convolutions followed by a max-pooling layer. They created a model that has an accuracy of 73.28 % on the FER2013 dataset.

A more intricate model uses the VGG19 (Simonyan & Zisserman, 2014) model and incorporates U-Net segmentation layers positioned between the layers of the VGG19 to accentuate additional features extracted from the feature map (Vignesh et al., 2023). This mechanism also regulates the passage of redundant information across the VGG layers. Vignesh et al. obtained 75.97% accuracy on the FER2013 dataset.

In facial emotion recognition, unsupervised learning approaches often involve clustering or dimensionality reduction techniques to group similar facial expressions together or to extract relevant features from the facial images. These methods can be particularly useful when labelled data is scarce or expensive to obtain, as they can potentially uncover hidden patterns or structures in the data without the need for manual annotation.

Xue, Sun & Yang (2023) investigated the application of unsupervised learning by exploring aspects such as scaling, annotation bias, and the disparity between discrete labels and continuous emotions. Through ContraWarping, they improved the pre-training of their model.

Yu, Sun & Yang (2022) took a distinct approach, employing CycleGAN and a Crossentropy loss function to produce neural expressions, which were subsequently combined to train CycleGAN. The classification utilised residual expression to generate neutral faces, thereby eliminating the labeling process.

3. Implementation and results

The objective of this paper is to investigate the development of a proficient model for facial emotion recognition using ChatGPT, presenting experiments tailored to improve the results. As stated before, the first step in the creation of the model is the choice of dataset. As mentioned earlier, the initial phase in developing the model involves selecting the dataset. In this study, the FER2013 dataset was utilized (Goodfellow et al., 2013), which is widely available and extensively used. Subsequently, ChatGPT was employed to inquire about aspects that could aid in establishing the base model for further investigation. As noted previously, the formulation of questions can impact the responses obtained. Therefore, a “Prompt Only” approach was opted, aligning with the objective of generating the base code using a zero-shot approach. Therefore, ChatGPT was challenged to generate models using both supervised and unsupervised learning.

3.1. ChatGPT with unsupervised learning

In this paper, the utilisation of unsupervised algorithms was considered as an alternative approach to address the challenges posed by the need for enhanced performance and extensive databases requiring laborious labelling and trained personnel.

The initial query prompted to ChatGPT was to "Create a FER model using unsupervised learning with the FER2013 dataset". It generated a model integrating StandardScaler for standardising data, PCA for reducing dimensionality and identifying primary and secondary components, and K-Means as the clustering algorithm. The resulting clusters are depicted in Figure 1, where happiness, fear, and surprise emerge as the most distinguishable emotions with minimal overlap. Angry, disgust, and neutral expressions exhibit intertwining, with neutral being particularly sensitive to misclassification.

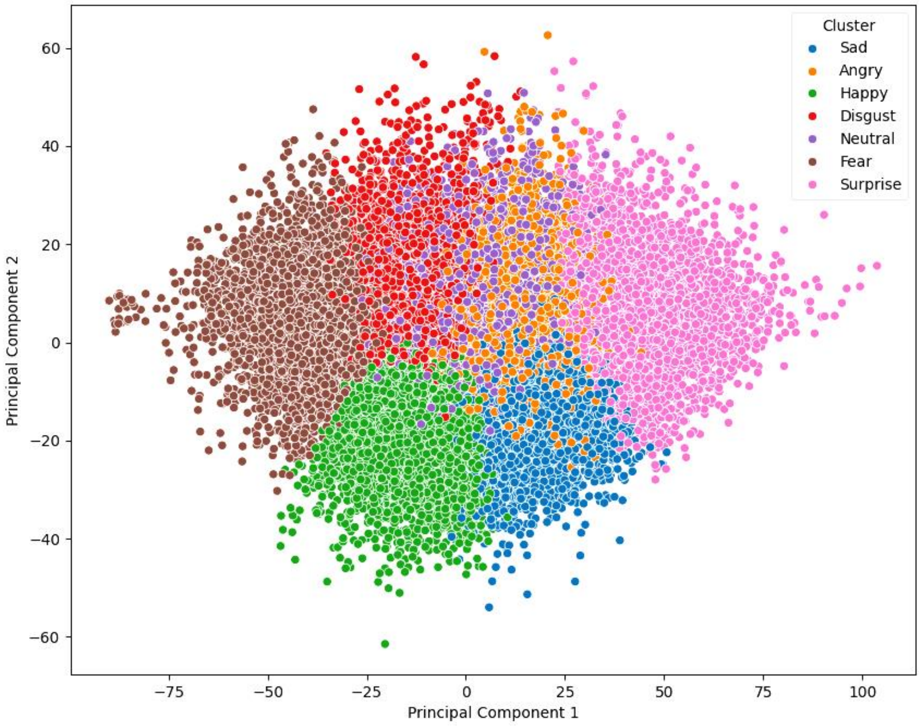

Figure 1. K-Means Cluster for FER2013

A second query posed to ChatGPT was to use a different algorithm that provided a similar model that used Gaussian Mixture Models (GMM) instead of K-means to group the labels together. Figure 2 shows a subpar performance for this clustering algorithm. While neutral appears predominantly in the upper portion of the graph, it is also misclassified across all quadrants. Similarly, happiness, primarily located in the third quadrant, is also distributed across other quadrants. The remaining emotions are sparsely scattered throughout the graph, lacking a discernible pattern.

The frequency of conversations with ChatGPT reached a juncture where the initial details shared in the earlier interactions became less relevant. The errors and the need to retry became even more pronounced. A new last approach was to use CNN as an unsupervised model. Although this practice is not common, it can be adapted to learn representations of the data without explicit labels. The proposed model by ChatGPT was three layers of convolution with max-pooling followed by a "flatten" layer, three dense layers, and a "reshape" layer to match the output to the input. As an optimiser, it used Adam, and for the loss, it used Mean Squared Error (MSE) Loss. After training the model on 10 epochs, the features are extracted in the second dense layer and provided as input for PCA. Observing this architecture, the model resembles an autoencoder, a type of neural network used for dimensionality reduction. In this case, the model is trained to reconstruct input images (the autoencoder's output matches its input) with a bottleneck layer in the middle that represents a compressed version of the input. Figure 3 shows a low performance with all the emotions scattered.

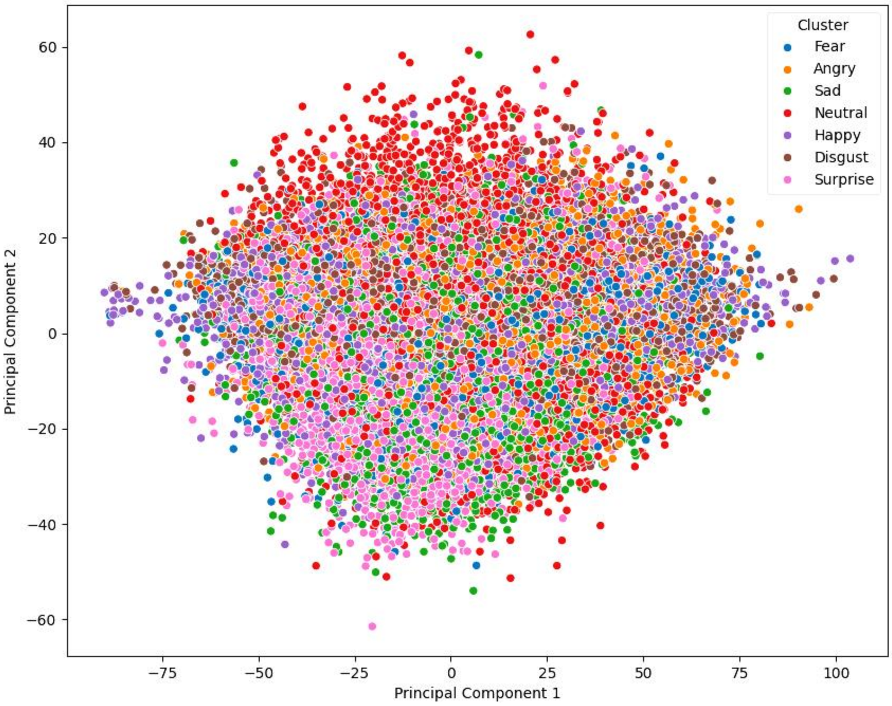

Figure 2. Gaussian Mixture Models Cluster for FER2013

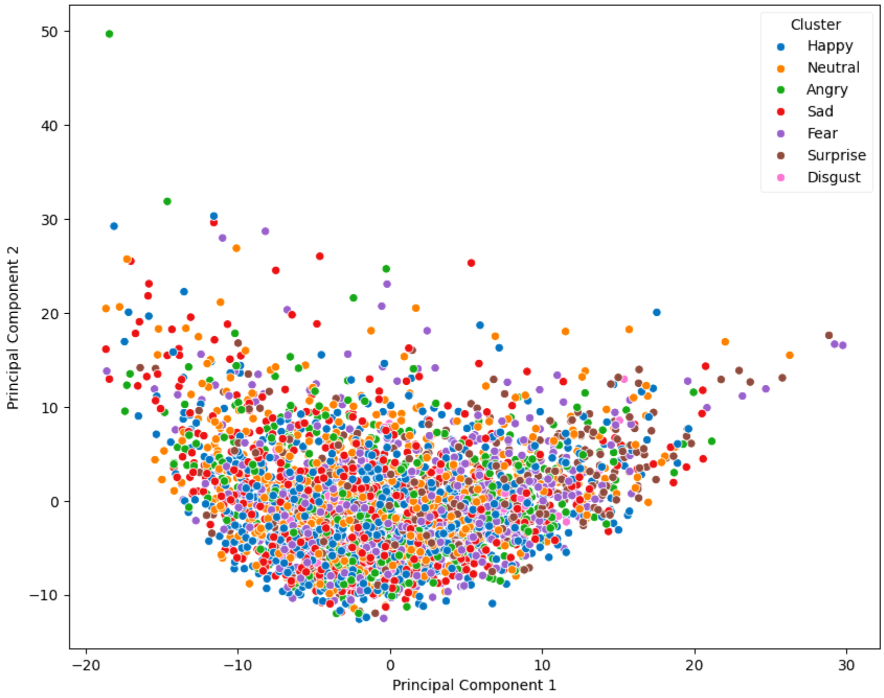

Figure 3. Unsupervised CNN for FER2013

As the generated models began to exhibit even poorer performance and errors became more frequent, we have chosen to conclude this research direction and transition to the supervised approach.

3.2. ChatGPT with supervised learning

The initial query posed was, "Create a ML model to recognise emotions in images using the FER2013 dataset". The resulting model consisted of three convolutional layers, each followed by a max pooling layer, and was trained for 10 epochs. Despite its simplistic design, ChatGPT3.5 provided all the necessary code to execute the model as specified. However, the achieved metrics were subpar, with an accuracy of 51.14%, and F1 scores of 69% for happiness and 66% for surprise. Extending the training to 30 epochs did not yield improvement, as the model appeared to suffer from underfitting due to its simplicity. A subsequent query to ChatGPT3.5 requested to "add more layers," prompting the addition of an extra set of convolutional layers and max pooling, bringing the total to four sets. Additionally, the number of training epochs was increased to 20. Despite these alterations, the model's performance remained comparable to the initial iteration, as evidenced by precision, recall, and F1-score metrics. Further inquiry to ChatGPT regarding the model's poor performance elicited theoretical insights into potential enhancements, including addressing issues such as low complexity, overfitting (despite indications of underfitting), data quality, hyperparameter sensitivity, class imbalance, evaluation metrics, and model initialisation.

Progressing beyond this stage presented challenges, as the insights received were largely theoretical and, though accurate in a general sense, not directly applicable to our specific scenario. In response, ChatGPT was requested to "please provide a better model." After several retries of refining the response, a different model was proposed. Notably, the conversation history stored in the language model (LLM) proved useful, eliminating the need to specify the dataset each time. This newly suggested model utilised the pre-trained VGG model (Simonyan & Zisserman, 2014) and leveraged transfer learning by incorporating additional final layers. However, executing the zero-shot output encountered hurdles, with one notable issue arising from the mismatch between the image channels expected by VGG (RGB format) and the single-channel grayscale images in FER2013. ChatGPT's proposed solution involved preprocessing the images to address this discrepancy. Another issue stemmed from the mismatch between the model's input shape expectations and the actual shape of the input data, which was resolved by removing redundant dimensions. Following these adjustments, the model was trained for 10 epochs, resulting in an accuracy of 39%, which increased to approximately 41% after 30 epochs. Subsequent discussions with ChatGPT aimed at addressing the low accuracy led to the exploration of various preprocessing strategies, including data augmentation, none of which yielded significant improvement. Additionally, there were iterations involving the VGG model as a base model, although it was ultimately not utilised in the final iteration.

The number of conversations with ChatGPT reached a point where the initial details conveyed in the earlier exchanges became less pertinent. Hence, an alternative strategy emerged: to provide comprehensive information in each inquiry, particularly focusing on the aspects previously overlooked or challenging for the model. This revised approach proved more effective, resulting in the generation of a new model comprising three convolutional layers followed by max-pooling and incorporating image preprocessing through data normalisation. This model achieved an accuracy of 53.94%. Furthermore, upon requesting ChatGPT to include an additional layer, it complied by adding both a convolutional and a max-pooling layer. Subsequently, the accuracy improved to 55.98% after 10 epochs, indicating progress in the model's design and performance.

Trying this new approach further, upon regeneration, a distinct model emerged. This latest model consists of four blocks, with the initial block comprising a convolutional layer, batch normalisation, and a RELU activation function to introduce non-linearity. The subsequent three layers consist of a residual block, max-pooling, and dropout. Within the residual block, two convolutional layers are present, along with batch normalisation and an activation layer, facilitating the feeding of the current layer's output to a deeper layer within the block. Trained on 30 epochs, it showed signs of overfitting, although this model achieves an accuracy of 62.49%, as can be seen in Table 1.

Table 1. Emotion Recognition Metrics (Residual Block Model)

Emotion	Precision	Recall	F1	Support
angry	0.54	0.57	0.55	985
disgust	0.69	0.55	0.61	102
fear	0.50	0.46	0.48	1043
happy	0.79	0.84	0.81	1765

sad	0.51	0.45	0.48	1210
surprise	0.76	0.75	0.75	795
neutral	0.56	0.60	0.58	1278
accuracy			0.62	7178
macro-average	0.62	0.60	0.61	7178
weighted average	0.62	0.62	0.62	7178

Using the same recent approach, a new model was obtained and trained for 20 epochs. The performance metrics can be seen in Table 2, where the accuracy stands at 60.08%. The model contains three blocks of layers, each block having two convolutional layers followed by batch normalisation, max pooling and dropout. Notably, this model stands out as one of the more intricate ones obtained. Happiness and surprise remain the most accurately recognised emotions, whereas detecting disgust poses challenges due to the limited available examples. Despite this, the detected emotions show consistency across the various analysed models.

Table 2. Emotion Recognition Metrics (Three Blocks Model)

Emotion	Precision	Recall	F1	Support
angry	0.50	0.59	0.54	985
disgust	0.70	0.48	0.57	102
fear	0.43	0.44	0.43	1043
happy	0.80	0.81	0.81	1765
sad	0.47	0.47	0.47	1210
surprise	0.71	0.74	0.72	795
neutral	0.60	0.51	0.55	1278
accuracy			0.60	7178
macro-average	0.60	0.58	0.58	7178
weighted average	0.60	0.60	0.60	7178

The accuracy may not match that of the model employing residual blocks. However, it proves to be a more suitable choice for the task, avoiding issues of overfitting or underfitting as shown in Figure 4 and Figure 5.

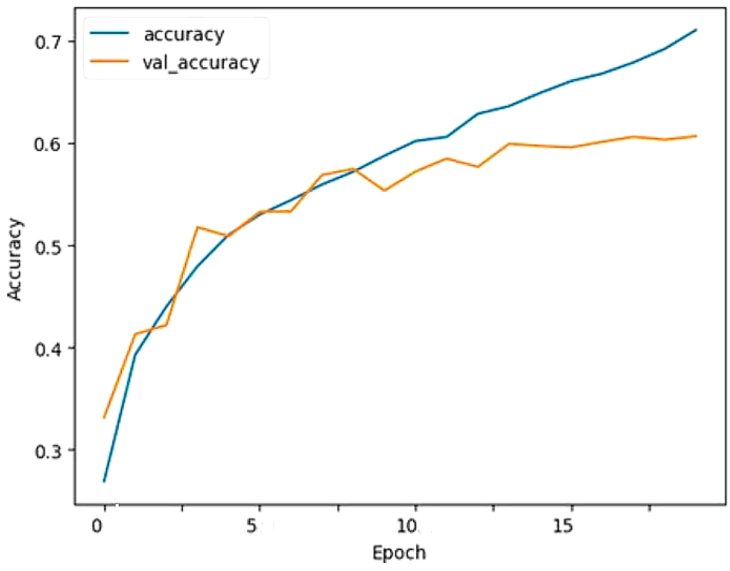

Figure 4. Three Blocks Model accuracy with 20 epochs

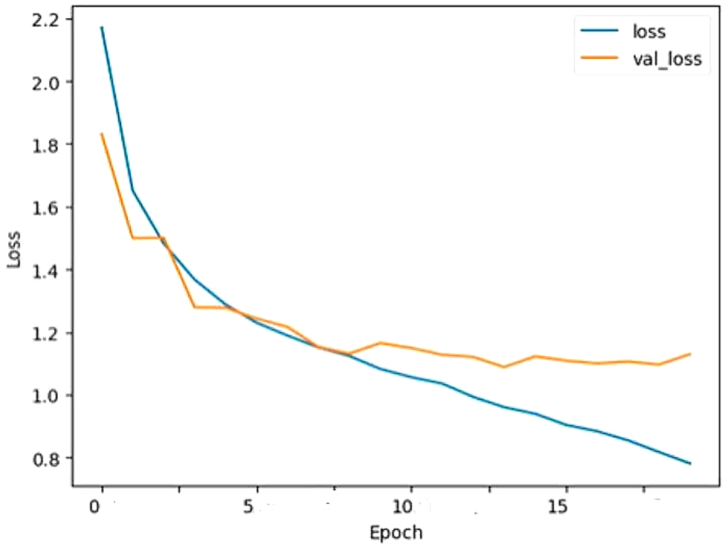

Figure 5. Three Blocks Model loss with 20 epochs

4. Discussions

The process of working with ChatGPT for model generation showcased both successes and challenges. However, due to the anticipated low accuracy and the substantial need for model improvement, employing ChatGPT in an unsupervised manner proves inefficient. Consequently, based on these findings, further investigation into this branch of machine learning was deemed unnecessary.

While various approaches were explored, including altering model architectures and preprocessing strategies, the results varied in terms of accuracy and code completeness. One such issue was preparing to use VGG16 by defining it as base model but ultimately not including it in the final model, as only three layers of convolutions were used instead. Despite encountering obstacles such as incomplete code outputs and models with low accuracy metrics, persistence and experimentation led to the development of increasingly complex and effective models.

Leveraging the conversational history with ChatGPT proved beneficial in refining the model generation process. Adjusting the approach to provide more comprehensive information in each interaction resulted in improved model designs and higher accuracies. Furthermore, the exploration of different model architectures, such as incorporating residual blocks inspired by models like ResNet, demonstrated the adaptability and versatility of the ChatGPT-assisted approach to model development.

In light of this, it can be inferred that, at present, leveraging a CNN model represents one of the most effective methods for achieving both high accuracy and significant reductions in research and development time. A noticeable discrepancy was observed in the utilisation of unsupervised compared to supervised methods, prompting an evaluation of ChatGPT's capacity to produce models. In the unsupervised setting, it generated fewer models of inferior quality with more errors, which is unsurprising given the limited examples available for training.

The iterative nature of the ChatGPT development process, combined with the need for experimentation and multiple interactions, has made the model development process less time-consuming than normal research. However, there are dependencies on prompting, as the effectiveness of ChatGPT's models heavily relied on the quality and specificity of the prompts, occasionally resulting in incomplete or inaccurate outputs. Additionally, ChatGPT's limited understanding of complex concepts and nuances in model development may have led to suboptimal suggestions or solutions.

The accuracy scores ranging from 60.08% to 62.49% achieved in facial emotion recognition represent an adequate achievement, particularly given the accelerated nature of model development. Despite the slightly lower accuracy compared to the state-of-the-art models reaching 73.28% to 75.97%, the advantage lies in the efficiency of model creation. The ability to quickly iterate and experiment with various configurations offers a distinct advantage, allowing for rapid exploration of different methodologies and approaches of facial emotion recognition systems. This method offers practical benefits in scenarios where time-to-market and development agility are desired, such as in rapidly evolving industries or research environments with tight deadlines. In essence, leveraging ChatGPT in an iterative model development approach demonstrated LLM's capacity to enhance efficiency and effectiveness in the pursuit of advanced AI applications like facial emotion recognition.

5. Conclusions

In conclusion, this study effectively investigates the utilisation of ChatGPT to expedite development processes and establish a solid foundation. Employing ChatGPT significantly reduces the time required for coding and debugging tasks, particularly beneficial at the outset of development and in offering suggestions for bug fixes, despite its current limitations.

The paper introduces an approach aimed at diminishing development duration while concurrently aiding in bug resolution.

A noticeable discrepancy was observed in the utilisation of unsupervised compared to supervised methods, prompting an evaluation of ChatGPT's capacity to produce models. In the unsupervised setting, it generated fewer models of inferior quality with more errors, which is unsurprising given the limited examples available for training although the model using K-Means performed well and should be investigated further.

Overall, while unsupervised learning techniques hold promise for facial emotion recognition, further research and development are needed to overcome challenges and fully leverage the potential of these methods in this domain.

For the supervised approach, experimental results underline the efficacy of swiftly crafting high-performing models with minimal time investment as a solution with 60.08% accuracy was obtained and validated in a matter of minutes. From this point on, the researcher has a good starting point to improve the model further and adjust it to the application needed.

This paper underlines the potential advantages of integrating ChatGPT into image processing workflows, notably in models pertaining to computer vision, graphics, and multimedia applications.

The findings show the role of word selection and developers' expertise in striking a balance between development speed and accuracy. As technological advancements progress, the prognostication of image processing workflows continues to improve, with this study providing invaluable insights into leveraging Large Language Models (LLMs) for expedited and efficient development processes.

This research presents the potential for future investigations into advanced LLMs, parameter optimisation, and their real-world applications, thus laying the groundwork for accelerated and agile image processing techniques.

REFERENCES

Ahmad, A., Waseem, M., Liang, P., Fehmideh, M., Aktar, M. S. & Mikkonen, T. (2023) Towards Human-Bot Collaborative Software Architecting with ChatGPT. To be published in *Software Engineering.* [Preprint] https://doi.org/10.48550/arXiv.2302.14600 [Accessed on March 2024].

Blocklove, J., Garg, S., Karri, R., & Pearce, H. (2023) Chip-Chat: Challenges and Opportunities in Conversational Hardware Design. To be published in *Machine Learning.* [Preprint] https://doi.org/10.48550/arXiv.2305.13243 [Accessed on April 2024].

Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Provost, M. E., Kim, S. K. & Narayanan, S. S. (2008) IEMOCAP: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*. 42(4), 335-359. doi:10.1007/s10579-008-9076-6.

Ekman, P. & Friesen, W. V. (1971) Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology.* 17(2), 124–129. doi:10.1037/h0030377.

Goodfellow, I. J. et al. (2013) Challenges in Representation Learning: A Report on Three Machine Learning Contests. In: Lee, M., Hirose, A., Hou, ZG., Kil, R.M. (eds.) Neural Information Processing, ICONIP 2013. *Lecture Notes in Computer Science,* 8228. Springer, Berlin, Heidelberg. pp.117-124. doi:10.1007/978-3-642-42051-1_16.

He, K., Zhang, X., Ren, S. & Sun, J. (2015) Deep Residual Learning for Image Recognition. To be published in *Computer Vision and Pattern Recognition.* [Preprint] https://doi.org/10.48550/arXiv.1512.03385 [Accessed on April 2024].

Khaireddin, Y. & Chen, Z. (2021) Facial Emotion Recognition: State of the Art Performance on FER2013. To be published in *Computer Vision and Pattern Recognition.* [Preprint] https://doi.org/10.48550/arXiv.2105.03588 [Accessed on March 2024].

Li, Y., Su, H., Shen, X., Li, W., Cao, Z. & Niu, S. (2017) DailyDialog: A Manually Labelled Multi-Turn Dialogue Dataset. To be published in *Computation and Language.* [Preprint] https://doi.org/10.48550/arXiv.1710.03957 [Accessed on April 2024].

Murr, L., Grainger, M. & Gao, D. (2023) Testing LLMs on Code Generation with Varying Levels of Prompt Specificity. To be published in *Software Engineering.* [Preprint] https://doi.org/10.48550/arXiv.2311.07599 [Accessed on April 2024].

OpenAI (2024) *ChatGPT*. https://openai.com/blog/chatgpt. [Acessed on March 2024].

Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E. & Mihalcea, R. (2018) Meld: A multimodal multi-party dataset for emotion recognition in conversations. To be published in *Computation and Language* [Preprint] https://doi.org/10.48550/arXiv.1810.02508 [Accessed on April 2024].

Simonyan, K. & Zisserman, A. (2014) Very Deep Convolutional Networks for Large-Scale Image Recognition.To published in *Computer Vision and Pattern Recognition.* [Preprint] https://doi.org/10.48550/arXiv.1409.1556 [Accessed on April 2024].

Tabachnyk, M. & Nikolov, S. (2022) *ML-Enhanced Code Completion Improves Developer Productivity.*: http://ai.googleblog.com/2022/07/ml-enhanced-code-completion-improves.html [Accessed in March 2024].

Vignesh, S., Savithadevi, M., Sridevi, M. & Sridhar, R. (2023) A novel facial emotion recognition model using segmentation VGG-19 architecture. *International Journal of Information Technology.* 15, 1777–1787. doi: 10.1007/s41870-023-01184-z.

Wake, N., Kanehira, A., Sasabuchi, K., Takamatsu, J. & Ikeuchi, K. (2023) Bias in Emotion Recognition with ChatGPT. To be published in *Robotics.* [Preprint] https://doi.org/10.48550/arXiv.2310.11753 [Accessed on April 2024].

Xue, F., Sun, Y. & Yang, Y. (2023) Unsupervised Facial Expression Representation Learning with Contrastive Local Warping. To be published in *Computer Vision and Pattern Recognition.* [Preprint] https://doi.org/10.48550/arXiv.2303.09034 [Accessed on April 2024].

Yu, Y., Sun, Y. & Yang, Z. (2022) An Unsupervised Facial Expression Recognition Method Based on CycleGAN. In *International Conference on Big Data, Information and Computer Network (BDICN)*, Sanya, China, 2022. pp. 669-674, doi: 10.1109/BDICN55575.2022.00129.

Zahiri, S. M. & Choi, J. D. (2018) Emotion Detection on TV Show Transcripts with Sequence-based Convolutional Neural Networks. The *Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence*. https://doi.org/10.48550/arXiv.1708.04299 [Accessed on April 2024].

Radu-Daniel BOLCAȘ graduated from the Faculty of Electronics, Telecommunications and Information Technology of the National University of Science and Technology Politehnica Bucharest, in 2017 and received his master's degree in the field of "Advanced Software Technologies in Communications" in 2019. He is currently a Ph.D. student in the field of Machine Learning at the same university. The main areas of interest are big data technologies, artificial intelligence, algorithms, and data structures.

Radu-Daniel BOLCAȘ a absolvit Facultatea de Electronică, Telecomunicații și Tehnologia Informației din cadrul Universității Naționale de Știință și Tehnologie Politehnica București, în anul 2017, iar studiile de masterat în domeniul „Tehnologii software avansate pentru comunicații", în anul 2019. În prezent este doctorand în cadrul aceleiași universități. Principalele sale domenii de interes sunt: tehnologiile Big Data, Data Mining, Inteligență Artificială, algoritmi și structuri de date.