International Journal of User-System Interaction 13(3) 2020, 121-135

Mobile Sentiment Analysis by Deep Learning Image Processing

Cristiana-Adriana Scăunașu, Paul Ștefan Popescu, Marian Cristian Mihăescu
University of Craiova
Department of Computers and Information Technology

Scaunasu.cristiana@gmail.com, {spopescu, mihaescu}@software.ucv.ro

Abstract. Once with the growth of free data sources which makes available more data for training machine learning and deep learning algorithms, new challenges tend to appear. Usage of deep learning algorithms for solving image classification problems became one of the most popular approaches because of its flexibility and generally good results. Sentiment analysis is a problem which was previously addressed using several approaches and for a different type of data which makes it a challenging problem. This paper presents a sentiment analysis solution for images gathered on mobile devices. Our approach uses deep learning for model training and part of the system validation and a mobile (android) application for model usage and real-time image classification. We considered the possibility to perform parameters tuning in order to obtain better performance, and we present a comparison between trained models. The experiments conducted with the trained models revealed excellent performance on the mobile device with excellent accuracy real-life tests.

Keywords: Deep learning, sentiment analysis, android application

DOI: 10.37789/ijusi.2020.13.3.1

1. Introduction

Image classification domain is a significant interest since a long time ago being used in many applications like object character recognition (OCR), face detection, medical applications, traffic sign recognition, etc. One particular usage of image classification is sentiment analysis because it analyses the human facial expression and can offer a good insight regarding their emotions. As a subsection of image classification, sentiment analysis faces many challenges Hussein (2018) like using it in social media Wang & Li (2015) or in some cases, even for health problems Yazdavar et al 2020.

Our approach refers to image sentiment analysis based on deep learning algorithms applied in images collected in real-time from the gallery of android mobile devices. For model training, we use TensorFlow Abadi et. al. (2016) deep learning library, which offers two essential benefits: flexibility and a lite model which can be run on mobile devices. Having such a possibility to run on a mobile device offers the advantaje to use the deep learning algorithm without having an internet connection and a server which can process and output the result. Classifying the images on a mobile device without a server-side application solves many problems because it offers excellent performance skipping the whole process of sending a frame, processing and then returning the classification label; this approach also allows user to performvery fast real-time classification.

One specific benefit of using the deep learning model on android devices is that android SDK facilitates developers to perform automatic face detection which is very useful for the topic of sentiment analysis especially when having more than one face in the image. This feature allows us to tag faces and apply the sentiment analysis algorithm on each one.

There are several functionalities integrated into the system presented in this paper which allows users to get the emotion probability in real-time from one of the cameras of their smartphone or gallery pictures based on one of the previously trained models. Using the Tensorflow lite converted model on the mobile device offers the user autonomy and makes the application usable even without a mobile connection. Another benefit is that having the model on the device offers higher performance than using a server for running the model.

The application is strongly related to behavioural sciences and Human-Computer Interaction research area allowing computers (which in our case is the mobile device) to understand the users better and get feedback by analyzing their reactions. Such a system can also be used to gain information is regarding the usability of an application because it can log the user's emotion and get relevant information about their emotions after completing specific actions. Another usage of this approach is in adaptive systems which can get automatic feedback from the user by analyzing its emotions and can adjust their interface in order to increase the usability. Part of the system presented in this paper refers to emotion recognition from images saved in the gallery, and this case can be used for better understanding the emotions and self-analysis.

One usage example for which the project presented in this paper was designed is to have a smart android device installed in the car with the camera facing the driver and providing feedback regarding their emotions. It is useful to have a device which can trigger a signal when several emotions are detected as these emotions can influence the driver's attention. For example, being angry or upset may influence the driver's reactions and cognitive functions, and this can put him on a risk. In such a situation, signalling this mood to the driver and recommending him to take a break until everything goes back to normal can firmly eliminate the risk of an accident. Further development can bring a functional integration with other multimedia devices installed on the vehicle, and the android application can bring several changes to the car music system or led colours/intensity (for cars that have light-controlled options installed). Still, in this usage scenario, it is essential to have the possibility to get the emotion for more than one person because even if the driver has a good mood. However, if another person from the car is in a bad mood, it may transfer this to the driver putting altogether at risk.

Regarding the system validation techniques we use both accuracy and standardized images for validation because even if we can obtain a good accuracy for trained images, we need to provide a generic model which can obtain good results on a wide variety of human faces.

2. Related Work

Sentiment analysis is a broad area Zhang et al. (2018) being implemented in various ways depending on what type of data is available and what methods are used. Because there is a lot of data available in text format, most of the sentiment analysis is performed on this type of data You et al. (2016), and further approaches refer opinion mining Hajmohammadi et. al (2012) or hybrid approaches Cai & Xia (2015). Other approaches for sentiment analysis refer to data collected from EEG sensors Prasad et al. (2018) or data collected from sensors of the mobile devices Kushawaha et al. (2020).

Our approach uses image classification using deep learning algorithms You et al. (2015) as the main approach. However, as input data, we use fer2013 dataset Giannopoulos et al. (2018), which is published on Kaggle, and there are several related papers which describes related approaches. One exciting approach Kumar & Jaiswal (2017), which shows improved performance for analyzing the sentiments associated with the images is to train and test on completely different datasets. In Kumar & Jaiswal (2017) authors trained a Convolutional Neural Network (CNN) on images extracted from Flickr platform (https://www.flickr.com) and then used images from a different platform (Twitter) for testing purposes.

Another paper published by Jinda & Singh (2015) that tackles the same problem presents an approach for image sentiment analysis using deep CNN with domain-specific fine-tuning. This framework is pre-trained on large-scale data for object recognition in order to perform transfer learning further. The authors used the same Flicker dataset as in Kumar & Jaiswal (2017), which was manually labelled and conducted extensive experiments on it. They used a progressive strategy of domain-specific fine-tuning of a deep neural network, and the results show that their proposed CNN can achieve better performance in image sentiment analysis than competing networks.

Boosting the sentiment analysis applied on images with visual attention is another innovative approach is Song et al. (2018) which aims to determine the attitude of a speaker or writer regarding a topic or the whole polarity of a document. The authors observed that the sentiment of images could be reflected only by some spatial regions, and they raise the question of how to locate the attended spatial areas for enhancing image sentiment analysis. As a response, they present a novel architecture that integrates visual attention into a successful CNN sentiment classification framework by training them into an end-to-end manner. They developed multiple layers to generate the attention distribution over the regions of the image. Furthermore, the saliency map of the selected image is employed as a priori knowledge to refine the attention distribution for sentiment prediction holistically. They conducted extensive experiments on Twitter and ARTphoto benchmarks, and their framework achieved better results compared to the state-of-the-art techniques.

Multimodal sentiment analysis is reviewed and summarized in Soleymani et al. (2017) as it offers promising avenues for analyzing facial and vocal expressions in addition to the transcript or textual content. This approach leverage emotion recognition and context inference to determine the underlying polarity and scope of an individual's sentiment. In the survey Soleymani et al. (2017), the authors define the problem of multimodal sentiment analysis and review it based on different domains, including image analysis. Still on the area of different approaches for images sentiment analysis is Sentribute by Yuan (2013), an image sentiment prediction framework, which leverages the mid-level attributes of an image to predict its sentiment. An empirical study of their proposed framework shows improved performance in terms of prediction accuracy. More importantly, by inspecting their prediction results, the authors can discover new relationships between mid-level attribute and sentiment encapsulated in the image.

Transfer learning approach used on visual sentiment analysis for social media images is another exciting approach by Islam & Zhang (2016), as the authors use hyper-parameters learned from a very deep CNN to initialize their network model in order to prevent it from overfitting. They conduct extensive experiments on a Twitter image dataset Velcin et al. (2014) and prove that our model achieves better performance than the current state-of-the-art.

In order to provide the best results for sentiment analysis, there are several methods used to improve the accuracy score. One of the first choices for improving the accuracy is to fine-tune the CNN for visual sentiment prediction as described in Campos et al. (2017) or in Campos et al. (2015) where they perform a comparison of several performance-boosting strategies and then achieves 6.1% absolute accuracy improvement over the previous state-of-the-art. Besides this approach, there are also two others which are less investigates; the first one refers low-resolution recognition as stated in Wang et al. (2016) or usage of very deep neural network for high-resolution images Kim et al. (2016). Regarding the real-time sentiment analysis, there is another paper Arriaga et al. (2017) which uses the same very used CNN trained on IMDB and Fer2013 datasets and their system was validated by its deployment on a Care-O-bot 3 robot used during RoboCup@Home competitions.

3. System Design

The system consists of two main components, the desktop core used for deep learning algorithm training and the second one, which represents the android application. The connection between these two components is made using the conversion of the TensorFlow trained model.

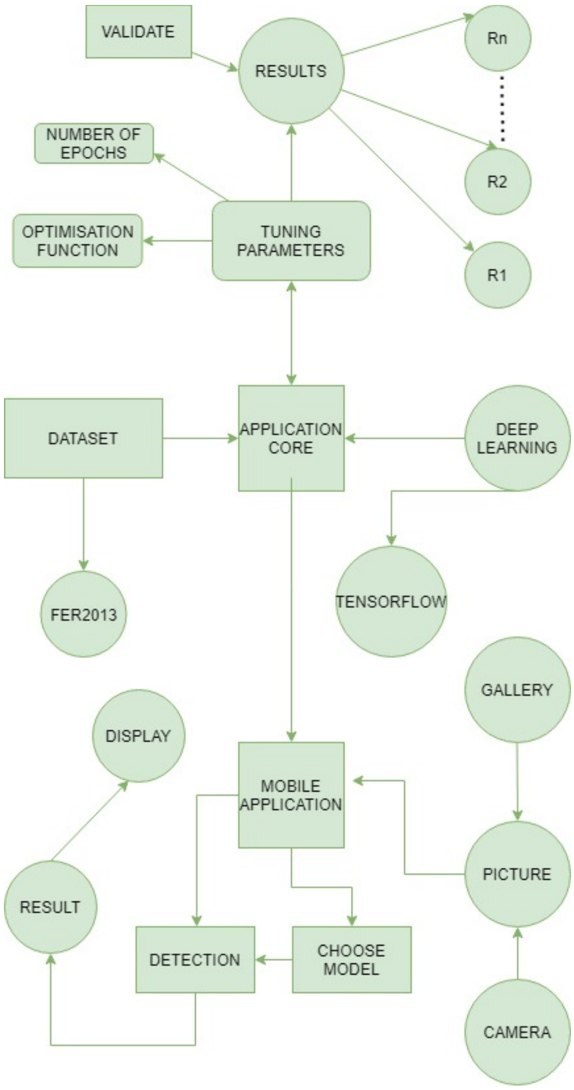

Figure 1. Overall System Design

Figure 1 presents the whole system flow and how the components interact with each other. In the centre of the figure, we have the application core, which in our case is the python script that is run via Google Colab and is used for model training. Because there are various tuning parameter configurations which can lead to different results; in order to optimize the model, we need to evaluate and retrain recursively. For results validations, tried different configurations of the number of epochs and the optimization function, and we aimed for the best results obtained for accuracy and loss function.

On the mobile application site, there is an android application which can compute and output the sentiment probability for a selected model. Carefully analyzing the probabilities can lead us to further model adjustments, and one of the implemented features is to be able to choose a model. This way, using real-life scenarios, we can select the most appropriate model.

Based on this approach, the application benefits from two-way validation: one using quality metrics (accuracy and loss function) an one using real-life scenarios on images imported from a gallery or gathered from the device camera.

1	emotion	pixels	Usage
2	0	70 80 82 72 58 58 60 63 54 58 60 48 89 115 121 119 115 110 98	Training
3	0	151 150 147 155 148 133 111 140 170 174 182 154 153 164 17	Training
4	2	231 212 156 164 174 138 161 173 182 200 106 38 39 74 138 16	Training
5	4	24 32 36 30 32 23 19 20 30 41 21 22 32 34 21 19 43 52 13 26 40	Training
6	6	4 0 0 0 0 0 0 0 0 0 0 0 3 15 23 28 48 50 58 84 115 127 137 142 1	Training
7	2	55 55 55 55 55 54 60 68 54 85 151 163 170 179 181 185 188 18	Training
8	4	20 17 19 21 25 38 42 42 46 54 56 62 63 66 82 108 118 130 139	Training
9	3	77 78 79 79 78 75 60 55 47 48 58 73 77 79 57 50 37 44 56 70 80	Training
10	3	85 84 90 121 101 102 133 153 153 169 177 189 195 199 205 20	Training
11	2	255 254 255 254 254 179 122 107 95 124 149 150 169 178 179	Training
12	0	30 24 21 23 25 25 49 67 84 103 120 125 130 139 140 139 148 1	Training
13	6	39 75 78 58 58 45 49 48 103 156 81 45 41 38 49 56 60 49 32 31	Training
14	6	219 213 206 202 209 217 216 215 219 218 223 230 227 227 23	Training
15	6	148 144 130 129 119 122 129 131 139 153 140 128 139 144 14	Training

Figure 2. Dataset example

Figure 2 presents a snip from the dataset used for training the neural network which is in .csv format, each line of it representing an image—every image from the dataset labelled from zero to six, each number representing a sentiment. The last column (usage) is used only for performance evaluation purposes and helps us to split between training and testing. The dataset consists of 35.888 de encoded images (32.298 for training 3.589 for testing). The pixels column represents a row of 2304 de numbers, each of them representing the values for pixel intensity from 0 to 255 where 0 corresponds to black colour and 255 to white.

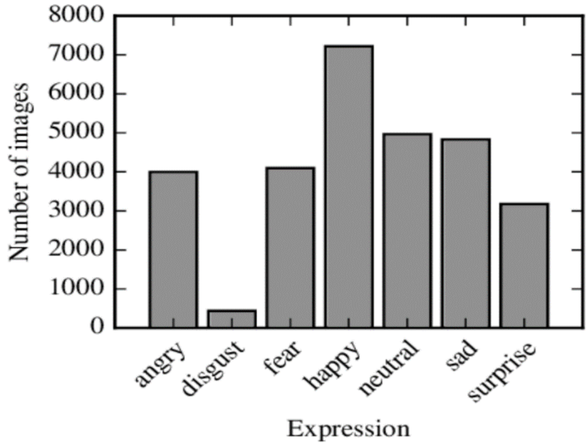

Figure 3. Sentiment distribution in the dataset

There are seven sentiments captured in the dataset with a happy state having most of the instances and disgust having the least number of instances. This proportion is essential because it influences model training and so the sentiment analysis results.

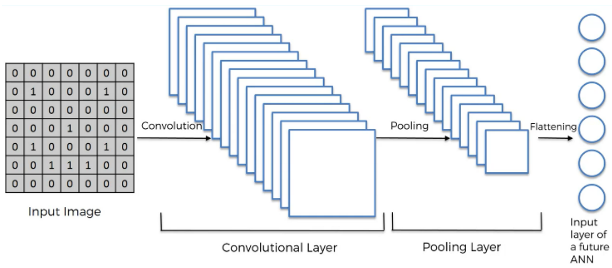

Figure 4. Feature preprocessing

Figure 4 presents the flow used for training the model. We use Keras to create a CNN, and the convolutional layer is used to compute the features, and then the pooling layer is used to reduce the input dimension. The last step is the image flattening which converts the data from matrices to vectors suitable for neural network input.

4. Experiments

The experiments conducted on the above-presented system are divided into two main categories: one for neural network validation and one for real-life testing on unseen data. This approach ensures us to provide a generic solution with excellent results which is not context-dependent because we need to ensure that the neural network does not learn features that apply only to some specific facial features.

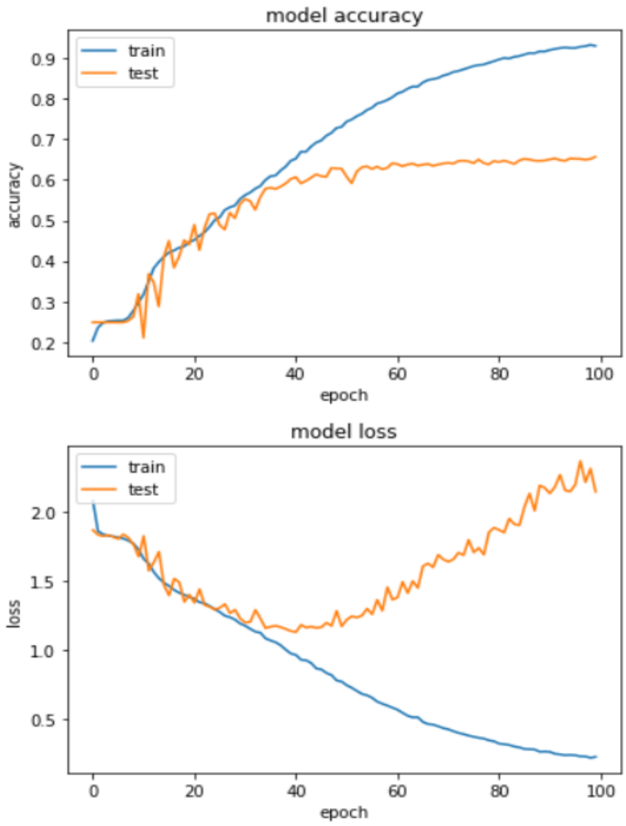

Figure 5. Model training

Figure 5 presents the accuracy and loss function evolution after training the model on one hundred epochs. On OX axes, we have the epochs number, and on OY we have the accuracy value and the loss function plotted for both train and test data. The model figure refers to model 1, which is discussed in the tables below. Going further than 100 epochs when training the model is not bringing significant benefits on the model performance. We choose to use both accuracy and loss function on train and data in order to get a better insight regarding the model performance because as we can see on the accuracy plot, even if the accuracy on the train goes above 0.9, on test data we get a plateau at slightly above 0.6. We need to mention that the accuracy on test dataset is relevant for real life scenarios as the application will be used on data which was not included in the training dataset.

The experiments conducted on the above-presented system are divided into two main categories: one for neural network validation and one for real-life testing on unseen data. This approach ensures us to provide a generic solution with good results which are not context-dependent because we need to ensure that the neural network does not learn features that apply only to some specific facial features.

Table 1. Model's description

Model	**Number of levels**	**Number of epochs**	**Learning rate**
M1	11	100	0.001
M2	11	1000	0.0001
M3	14	100	0.001

Table 1 presents the tuning parameters which lead us to the best results. On the first column, we have the models and on the next, the number of levels within the network, the number of epochs we trained and then, in the last column, the learning rate.

Table 2. Accuracy on train test and time spent

Model	**Time**	**Test**	**Train**
M1	1h and 8min	0.65	0.65
M2	6h and 30min	0.62	0.64
M3	3h and 20min	0.61	0.60

In table 2 we can see the time spent on training and then the accuracy obtained for train and tests datasets. This table is related to Table 1 which means that the first model (M1) had 11 levels, was trained on 100 epochs with a learning rate of 0.001 and took one hour and eight minutes to train, obtaining a 0.65 (out of 1) for the test dataset and similar accuracy of 0.65 on the training dataset. One thing that needs to be mentioned is that the test data was not included in the train data so it is treated as unseen data. Regarding the trained models, the accuracy for both training and testing are very close even if we trained the model over a different amount of epochs and with different learning rates which implied a significant increase in training time. The conclusion is that we reached a plateau so further training will not bring significant benefits.

Table 3. Accuracy for each sentiment

Model	**0**	**1**	**2**	**3**	**4**	**5**	**6**
M1	1.00	1.00	0.75	0.66	0.94	0.87	1.00
M2	0.97	0.96	0.15	0.77	0.96	0.61	0.83
M3	1.00	0.47	-	0.83	0.83	0.62	0.71

Table 3 presents the results obtained for computing each of the sentiments included in the datasets. From 0 to 6, the values from the first row corresponds to one sentiment: 0 for *happy*, 1 for *sad*, 2 for *disgust*, 3 for *neutral*, 4 four *angry*, 5 for *fear* and six for *surprise*.

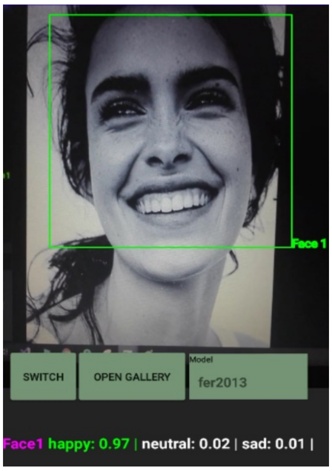

Figure 6. Results for one person

The results are not very well balanced because the data available for each sentiment differs, and as we can see in figure 3 for disgust, we have the least amount of data, and this corresponds to the worth results. Some of the classes like happy or sad benefits from the significant amount of data and brings us an excellent classification score. Regarding the values which are in a range from 0 to 1, this value represents the confidence of the model regarding that state. For example, for the first column and the first model, we choose a picture which have a happy face and model M1 detects this emotion with a confidence of 1.00 which is the maximum value, M2 very close with a confidence of 0.97 and M3 also with 1.00.

Figure 6 presents a short example of sentiment analysis using the smartphone camera, pointing to a single person. First, we detect the face in the image, and then we select the three best probabilities for the emotion in the picture. If the accuracy for one sentiment exceeds 0.3, we change the colour of the sentiment in green as we consider it is relevant enough. The motivation for printing the most relevant three emotion instead of the one with the best probability is that some the face scanned may produce very similar probabilities for two emotions.

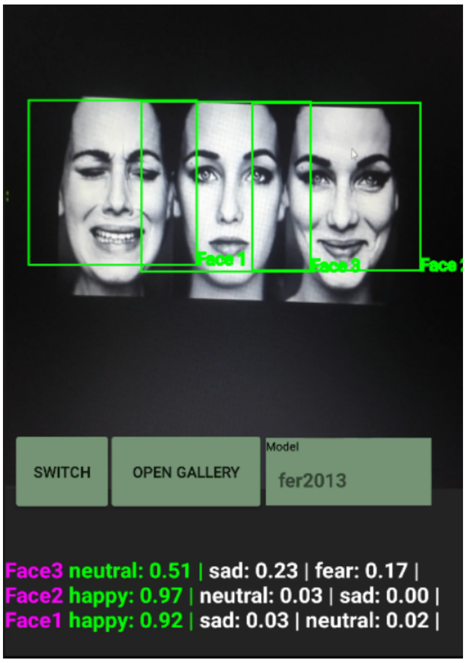

Figure 7. Results on multiple faces

Figure 7 presents the situation in which we have multiple faces in the same picture or image gathered from the smartphone camera. In this case, we also detect the faces and tag them so we can see which emotion is in each face. As we can see in the printscreen we also tag the images and we print the probabilities for the emotions detected for each face.

5. Conclusion

This paper presents a sentiment analysis approach using deep learning algorithms implemented for mobile devices. Our approach used TensorFlow with Keras for model training and tuning and android SDK for mobile application development. Besides the application development and system validation, we investigated if there are possibilities to improve the system's performance and accuracy by adjusting some parameters of the trained model. We also considered a pre-trained model in order to evaluate the possibility of using it, but we could not get significantly better results.

As future work, we plan to improve the model adding more data to the dataset and also using other datasets because one of the problems we faced when training the data was that some classes did not have enough data for proper training. For evaluation purposes, in our case, we considered a limited number of pictures, but there are far more, which can be used to get a better insight regarding the model.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Zheng, X. (2016). Tensorflow: A system for large-scale machine learning. In 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16) (pp. 265-283).

Arriaga, O., Valdenegro-Toro, M., & Plöger, P. (2017). Real-time convolutional neural networks for emotion and gender classification. arXiv preprint arXiv:1710.07557.

Cai, G., & Xia, B. (2015). Convolutional neural networks for multimedia sentiment analysis. In Natural Language Processing and Chinese Computing (pp. 159-167). Springer, Cham. https://doi.org/10.1007/978-3-319-25207-0_14.

Campos, V., Jou, B., & Giro-i-Nieto, X. (2017). From pixels to sentiment: Fine-tuning CNNs for visual sentiment prediction. Image and Vision Computing, 65, 15-22. https://doi.org/10.1016/j.imavis.2017.01.011.

Campos, V., Salvador, A., Giró-i-Nieto, X., & Jou, B. (2015, October). Diving deep into sentiment: Understanding fine-tuned CNNs for visual sentiment prediction. In

Proceedings of the 1st International Workshop on Affect & Sentiment in Multimedia (pp. 57-62). https://doi.org/10.1145/2813524.2813530.

Giannopoulos, P., Perikos, I., & Hatzilygeroudis, I. (2018). Deep learning approaches for facial emotion recognition: A case study on FER-2013. In Advances in hybridization of intelligent methods (pp. 1-16). Springer, Cham. https://doi.org/10.1007/978-3-319-66790-4_1.

Hajmohammadi MS, Ibrahim R, Othman ZA. (2012) Opinion mining and sentiment analysis: A survey. International Journal of Computers & Technology. 2012 Jun 30;2(3):171-8.

Hussein, D. M. E. D. M. (2018). A survey on sentiment analysis challenges. Journal of King Saud University-Engineering Sciences, 30(4), 330-338. https://doi.org/10.1016/j.jksues.2016.04.002.

Islam, J., & Zhang, Y. (2016, October). Visual sentiment analysis for social images using transfer learning approach. In 2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)(BDCloud-SocialCom-SustainCom) (pp. 124-130). IEEE. 10.1109/BDCloud-SocialCom-SustainCom.2016.29.

Jindal, S., & Singh, S. (2015, December). Image sentiment analysis using deep convolutional neural networks with domain specific fine tuning. In 2015 International Conference on Information Processing (ICIP) (pp. 447-451). IEEE., 10.1109/INFOP.2015.7489424.

Kim, J., Lee, J. K., & Lee, K. M. (2016). Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1646-1654).

Kumar, A., & Jaiswal, A. (2017, December). Image sentiment analysis using convolutional neural network. In International Conference on Intelligent Systems Design and Applications (pp. 464-473). Springer, Cham. https://doi.org/10.1007/978-3-319-76348-4_45.

Kushawaha, D., De, D., Mohindru, V., & Gupta, A. K. (2020). Sentiment analysis and mood detection on an Android platform using machine learning integrated with Internet of Things. In Proceedings of ICRIC 2019 (pp. 223-238). Springer, Cham. https://doi.org/10.1007/978-3-030-29407-6_18.

Prasad, D. K., Liu, S., Chen, S. H. A., & Quek, C. (2018). Sentiment analysis using EEG activities for suicidology. Expert Systems with Applications, 103, 206-217. https://doi.org/10.1016/j.eswa.2018.03.01.

Soleymani, M., Garcia, D., Jou, B., Schuller, B., Chang, S. F., & Pantic, M. (2017). A survey of multimodal sentiment analysis. Image and Vision Computing, 65, 3-14. https://doi.org/10.1016/j.imavis.2017.08.003.

Song, K., Yao, T., Ling, Q., & Mei, T. (2018). Boosting image sentiment analysis with visual attention. Neurocomputing, 312, 218-228. https://doi.org/10.1016/j.neucom.2018.05.104.

Velcin, J., Brun, C., Dormagen, J. Y., Kim, Y. M., Roux, C., Boyadjian, J., ... & Molina, A. (2014, May). Investigating the image of entities in social media: Dataset design and first results. In 9th International Conference on Language Resources and Evaluation..

Wang, Y., & Li, B. (2015, November). Sentiment analysis for social media images. In 2015 IEEE international conference on data mining workshop (ICDMW) (pp. 1584-1591). IEEE. DOI: 10.1109/ICDMW.2015.142.

Wang, Z., Chang, S., Yang, Y., Liu, D., & Huang, T. S. (2016). Studying very low resolution recognition using deep networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4792-4800).

Yazdavar, A. H., Mahdavinejad, M. S., Bajaj, G., Romine, W., Sheth, A., Monadjemi, A. H., ... & Hitzler, P. (2020). Multimodal mental health analysis in social media. Plos one, 15(4), e0226248. https://doi.org/10.1371/journal.pone.0226248.

.