Sigma J Eng Nat Sci, Vol. 42, No. 4, pp. 1133–1147, August, 2024

Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2024.00093

Research Article

Blocking harmful images with a deep learning based next generation firewall

Kenan BAYSAL[1,*], Deniz TAŞKIN[2]

[1]*Hayrabolu Vocational School, Tekirdag Namik Kemal University, Tekirdag, 59400, Türkiye*
[2]*Department of Computer Engineering, Trakya University, Edirne, 22030, Türkiye*

ARTICLE INFO

Article history
Received: 06 January 2023
Revised: 16 February 2023
Accepted: 12 March 2023

Keywords:
Convolutional neural network; Deep learning; Firewall

ABSTRACT

There are various blocking and filtering algorithms for protection against harmful contents on the Internet. However, it is impossible to classify particularly the visual contents according to their genres and block them through traditional methods. In order to block the harmful visual contents, such as various advertisements and social media posts, we need to review and classify them as per their contents. Deep learning method is today's most efficient method to review the visual contents.

In this study, only the harmful images were blocked without completely blocking the entire website. Alcoholic drinks were selected as the harmful content data set. For this purpose, a training was provided with 4.6 million images by using CNN (Convolutional Neural Networks) and GoogLeNet architecture. At the end of this training, 97.6469% of accuracy was achieved. F1 score was calculated as 87.75526188% at the end of the test conducted with 154501 images.

The images were determined through the network traffic via mitmproxy and classified as harmful or harmless thanks to the trained model, and the filtering process was successfully completed.

Cite this article as: Baysal K, Taşkın D. Blocking harmful images with a deep learning based next generation firewall. Sigma J Eng Nat Sci 2024;42(4):1133–1147.

INTRODUCTION

Artificial neural networks aim to create machines that can decide on their own with living mathematical models inspired by biological neural tissue samples. The idea of creating a mathematically similar model to the biological structure of the brain in a computer environment was suggested by McCulloch et al. [1] They showed that logical expressions can be created with the artificial nerve cell they created using logical electrical circuits. It is accepted that the first artificial nerve cell was created with this study. It was concluded that more than one artificial nerve cell should be used to create a useful structure, and the foundation of artificial neural networks was laid out.

Nowadays, various studies are carried out in many fields by making use of artificial neural networks. Demir and Karaboga have studied on successful and unsuccessful

***Corresponding author.**
*E-mail address: kbaysal@nku.edu.tr

This paper was recommended for publication in revised form by Editor in-Chief Ahmet Selim Dalkilic

students mathematical achievements [2]. According to their studies results, Jordan Neural Networks (JNN) has been more successful to predict students achievements than the others. Rajkovic et al have used ANNs in their study to optimize parameters of producing biodiesel from sunflower oil [3]. Zettler et al have used ANN combined with fuzzy logic for pressure sensitive grouting on rock surfaces [4]. They used ANNs on Transient Pressure Analysis (TPA) section to control grouting process. Ma et al have used ANNs to early detection and diagnosis of chronical kidney disease [5]. They used Heterogeneous Modified Artificial Neural Network (HMANN) method, which is reduced the noise and helps a clear identification on kidney images, for training the model. Sabilla et al have worked on an electronic nose on their studies [6]. They have used ANNs for estimate gas concentration. They used fresh air, raw mango, ripe mango and rotten mango parameters to train the model. According to their results model has good results on prediction. Esen et al. have used ANNs modelling a solar assisted heat pump systems [7]. And they have made performance analysis by using ANNs. Esen et al also studied to predicting performance of a ground-source heat pump systems using adaptive-network based fuzzy inference systems (ANFIS) [8]. According to their conclusion, ANFIS can be used to predict performance for such ground-source heat pump systems. Efe and Alganci have used machine learning methods to classify the land covers from satellite images [9]. They classified the images as, artificial fields, forests, water bodies, bare and semi-natural areas, agricultural fields and urban texture. They used these classified data to detect urban expansion in the areas they studied.

The Aim of the Study

Due to the high-speed development of electronics and computer technologies from the second half of the 1900s, the 2000s are known as the "Information Age" or the "Internet Age". With the rapid development and evolution of the concept of social media within the last decade, the use of Internet had observable and permanent impacts on human relations. Due to the Internet's inherently global structure, it is observed that activities, such as consumption, communication, and interaction, influence large masses in different ways regardless of space and time. Visual images constitute the majority of this interaction. According to 2021 report of Wearesocial.com website, there are 4.20 billion active social media users among 4.66 billion active Internet users throughout the world. According to the same data, people spend approximately 6 hours and 54 minutes daily on devices connected to the Internet. In Turkey, people spend approximately 2 hours and 57 minutes daily on social media [10].

Cited from Lehu and Bressoud, Yaraş et al., stated that with the decreasing impact of traditional media, the advertisers began to use product placement technique as a marketing method [11]. Yaraş et al., stated that the companies prefer product placement techniques in order to find a way around advertisement restrictions for weapons, alcoholic drinks and tobacco products and influence particularly the young population [11].

A considerable part of the contents posted on social media includes product promotions and product placement. Although some contents, such as violence and sexuality, are blocked by the social media platforms, alcoholic drinks, tobacco use, or firearms are not subject to this filtering. The websites other than the social media contain more images that may affect the cognitive activities of the youth, such as violence and sexuality.

Especially the children are more vulnerable than the adults against the inappropriate contents of the websites. İplikçi and Batu conducted a content analysis study on advertisements on children's websites in Turkey. The results of this study showed that there were 18+ adult contents within these advertisements, and some other contents that might have a negative impact on children's mental health. Cited from Kapferer, İplikçi et al., stated that children under 5 years of age do not have any knowledge about the functions of advertisement images, and 26% of children consider advertisements as reliable sources of information [12].

There are numerous standards and regulations regarding visual media contents. In his study about media ethics and codes of conduct for children, Uzun, reviewed the ethical rules of BBC, however, he reported that considering the commercial structures of these web-casting organizations, they could not be expected to make any regulations regarding regulation and enhancement of these ethical rules. In this case, one may conclude that blocking and banning an entire website may fix this problem [13].

In his study on the impacts of visual media on children's mental health, Kanbur, stated that in today's world, completely prohibiting the children and the young from using visual media tools is not an effective solution. These restrictions may cause rejection and peer pressure among the social circle of the children. This may cause pressure on children [14].

It would not be a realistic approach to expect an enhancement and regulation on existing conditions from content creators, social media companies, and other digital media organizations regarding these contents. Therefore, in order to avoid these harmful contents, the users need filtering software programs. The filtering processes through the existing firewall system are performed as creating certain rules and word-based or address blocking. This system cannot detect whether there are inappropriate contents within an image. Thus, content-based filtering is not possible through traditional methods. Filtering process occurs in accordance with a predetermined domain list or a certain URL. This means complete blocking of a website. However, completely banning these channels, such as social media, which are home to various contents and environments, and which have a commercial value in terms of advertisement revenues, would lead to various problems. Therefore, content-based filtering is required.

Within this context, the purpose of this study is to detect contents through a smart algorithm, and to ensure safe Internet use without complete prohibition of the Internet.

Deep Learning

The concept of deep learning has been formed with contribution of various disciplines in the background as a subfield of machine learning. The first actually operating algorithm for deep learning is accepted as the study of Ivakhnenko and Lapa [15]. Although deep learning has come to a deadlock in the 90s due to hardware equipment inadequacy, with the increasing GPU performance, it was begun to be used in many fields within the last decade. Successful results were obtained via deep learning, such as computer games, speech and emotion recognition, natural language processing, and autonomous vehicles [16-19].

Deep learning is considered as a field of research with highest potential that will have a big impact for years to come [20]. CAFFE (Convolutional Architecture for Fast Feature Embedding), Tensorflow, Torch, and Keras are among the widely-used deep learning library stacks. Thanks to these open-source and easily-accessible library stacks, successful results were obtained in visual recognition and classification [21].

Deep learning uses the layer structure of artificial neural networks. It mimics the learning processes of human brain with mathematical models and processes the data and creates semantic concepts [22]. This system is called deep learning because the layers are interconnected, and the system has a multilayer and deep structure. It is accepted as a subfield of machine learning. Unlike machine learning, numerous hidden layers are used in artificial neural network layer structure of deep learning. The training is performed with high amount of data without deducing any features.

The algorithm, which is trained by using existing data sets, creates its own rules through multilayer artificial neural networks and can filter out the required information. Unlike traditional methods, training of deep learning algorithm requires large amounts of data sets. As the size of the data set increases, the working accuracy and efficiency of the trained algorithm increases [23]. As shown in Figure 1, the size of the neural network increases with the size of the data, and this has a positive impact on performance. Although a small neural network gives better results than traditional classifiers on a small amount of data, the large-scale neural networks provide higher accuracy performance on large amount of data [24].

Deep learning methods might not be always a best option. For example, Yiğit et al. have conducted studies in the financial field and compared traditional methods with deep learning methods to predict market price movements. In their study, they chose BIST30, BIST50 and BIST100 data as the data set. According to their results ARIMA method, which is mentioned as traditional method on their studies, has been better results then LSTM and GRU, which are deep learning methods. They referred that, deep learning methods has much better results for massive data sets, but ARIMA is one-step ahead forecasting on univariate datasets [25].

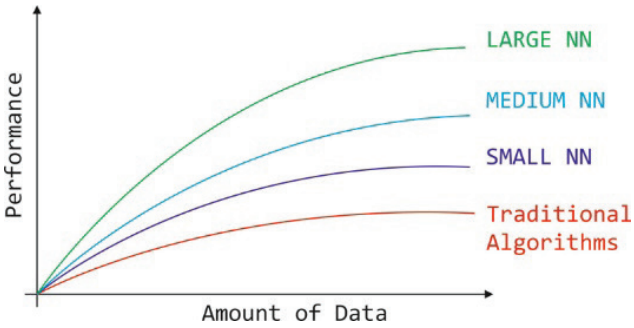


Figure 1. The relationship between data size and performance of deep neural networks [24].

Convolutional Neural Networks

Convolutional neural networks (CNN) are Artificial Neural Network (ANN) architectures developed based on the image processing method of the human eye. It is used for processes, such as filtering out, classification, or object recognition of shapes or scenes searched within the images. It learns about low-level characteristics of an object in order to describe it and learns about separating that object from the others. For example, in order to describe a cat, all sub-features that makes up a cat are gradually deduced. The large number of parameters calculated in traditional artificial neural network architectures lead to higher process power.

Expansion of data sets necessitates deduction of features particularly in processes, such as visual recognition. The image, which is the input value in CNN structure, are subjected to various filters in the convolution layer and the features are deducted, and the network is trained over these features. These filters may be various vertical and horizontal side filters, which will ensure side capturing in the image.

In their study, LeCun et al. became successful in recognizing hand-written numbers by using CNN architecture [26]. This study is considered as a pioneering work in CNN. After that, there has been a huge gap until the 2010 for studies on CNN. In due course, with the cell phone cameras becoming widespread, a boom of visual materials and tagged data has occurred in social media and other web environments, and the required effloresce has reached the desired level with GPU power. The deep learning model created by using CNN architecture has won the 2012 ImageNet image recognition competition, and CNN has been proven to be an efficient architecture [27]. In the long view, CNN architecture has become a popular method for image recognition and classification.

CNN training process is a multilayer structure. Throughout this training, the values obtained at the end

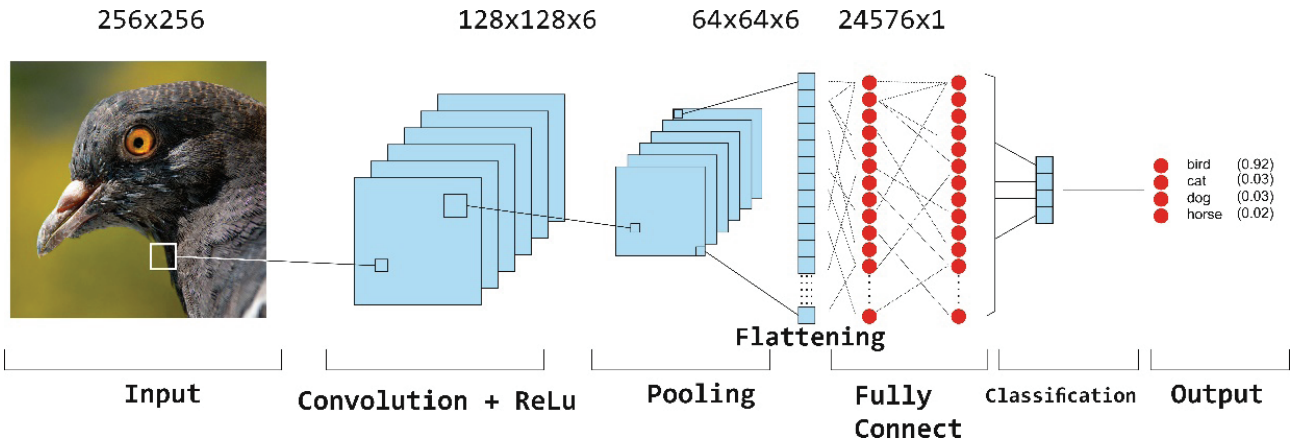


Figure 2. CNN Layer Structure.

of convolutional processes applied through various filters are subjected to activation functions. The backpropagation occurs and the weights are updated with the loss value obtained at the end of a training tour performed on neural networks. The training process continues until the best possible result is obtained.

The layers of CNN can be listed as the input layer, convolution layer (convolution + ReLU), pooling layer, full connection layer, classification layer, and output layer. The rankings of these layers are given in Figure 2.

The input layer is the first layer of the network, and the raw data is given to the network at this layer. If the size of the input data is too large at this stage, the time needed for training and the need for memory would increase. On the contrary, if the size of the input data is too small, the depth of the network would decrease since details would get lost, and the features that may be deduced would be insufficient. The size of the input data determines the depth of the network, the number of filters to be applied determines the width of the network.

MATERIALS AND METHODS

This study aims to develop a system to filter only the images instead of blocking an entire web site URL, where potentially harmful images may be present. A website may not always contain only harmful images. For example, a blog may prepare a content including these harmful images in only a few of its posts. The remaining contents may include images that may be helpful, rather than including harmful images. Social media is another example of this case. Social media platforms, such as Instagram, Facebook, Twitter, etc. may include helpful groups, lists, and posts of friends and family, but they may also include posts with images that can easily get through the filters of these sites. Completely blocking these websites, which are a part of social life in terms of communication, could escalate this problem to another level. In addition, URL-based blocking contains various challenges within itself. The blocked websites may get through this blocking with another address. Continuously tracking these changes and updating this blocked list can be considered as another workload. The filters of these websites can be flexible in terms of violence, sexuality, and substance abuse. In this study, the factors encouraging alcohol use were discussed, and a deep learning model was trained, which can recognize the images including alcoholic drinks and environments containing alcoholic drinks.

The process steps of this study Are given in Figure 3. This study consists of six main sections as provision of images for the data set, organization and tagging of the data set, selection of hyper-parameters, model performance evaluation, and firewall application.

The main harmful content selected for the filtering process was alcoholic beverages. The alcoholic drinks are gathered under 43 different titles solely according to their raw materials under the alcoholic drinks title of Wikipedia website. The same website lists fermented drinks under 64 titles and distilled drinks under 78 titles [28]. Considering each category has its own brands and packaging style, the size of the data set has to be extremely large. In addition, in case the color and packages of these drinks resemble the packages and color of nonalcoholic drinks, data for comparison with nonalcoholic beverages should be given in higher amounts in the data set. In addition to this variety, since there is a pattern resemblance between some alcoholic drinks and their real-life images, the real-life images were also included within the data set. At the image evaluation stage, the existence of alcoholic drinks within an image has also been considered rather than only blocking the packages of alcoholic drink images.

Data Set

Since there are no public data sets for alcoholic drinks at a size fit for this purpose, the images used in model training were obtained through the Internet via a web crawler designed in Python environment. The designed crawler structure worked continuously on an address database

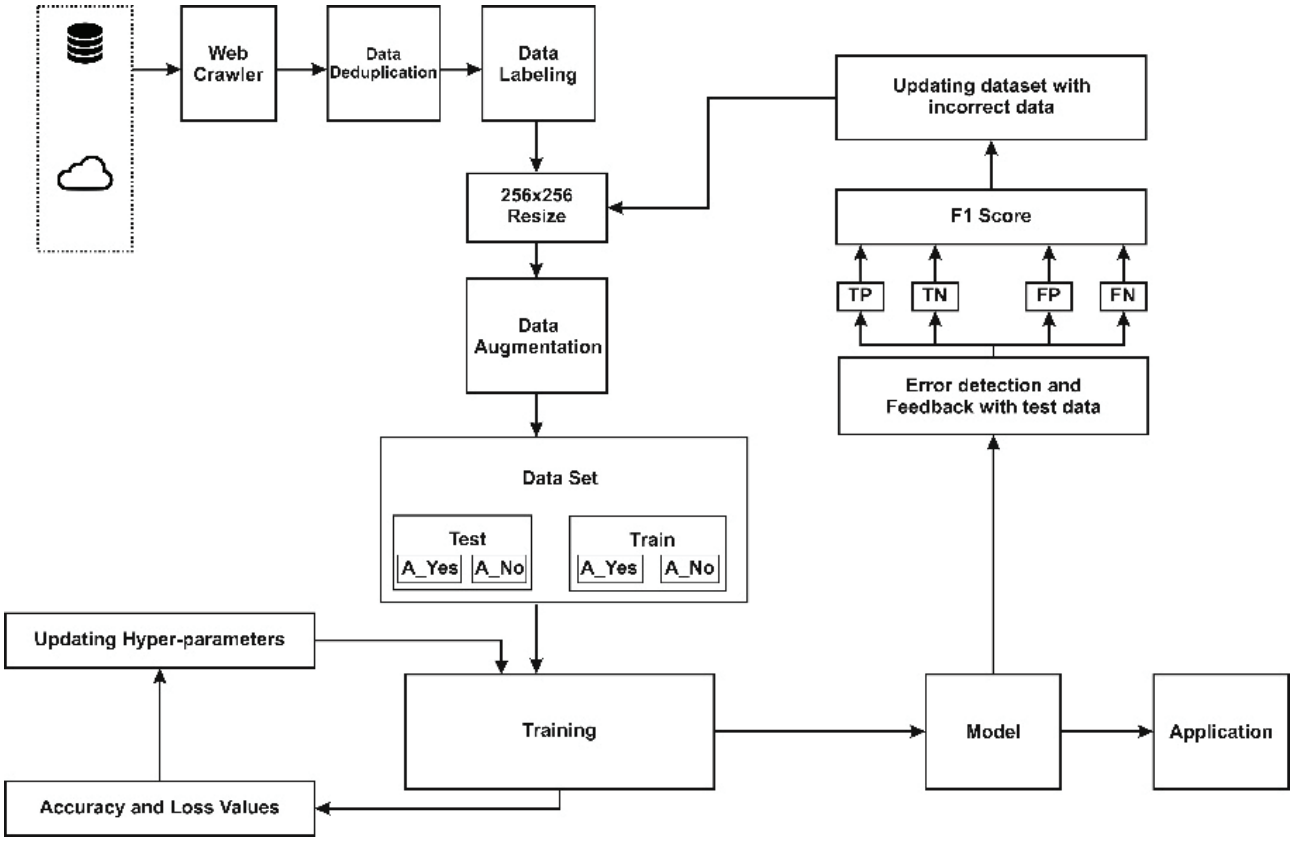


Figure 3. Process Steps.

containing various web addresses for 7 months, and approximately 3 million images were obtained.

Data sets were tagged in two main categories. The tags were determined as "alkol_var" which means "alcohol_yes" and "alkol_yok" which means "alcohol_no" according to whether there are alcoholic drinks within the image for filtering. Figure 4 shows a small example of the data set containing images tagged "alkol_var" and "alkol_yok".

While preparing this data set, not only the images containing alcoholic drinks in the foreground. Many different environments and scenes including exchange of toasts, crowded night clubs and entertainment venues that serve alcoholic drinks, and dinner tables including alcoholic drinks were tagged as "alkol_var".

Data Deduplication

In case an image is found more than once in the data set, the trained model may perform false learning towards a certain direction or begin to memorize. Therefore, all images were compared through Mean Squared Error (MSE) function. MSE is a risk function, and it is also used to measure estimation performance of machine learning models. It compares the desired value and the obtained value and presents a result. It always produces a positive value [29]. Thus, it is suitable for comparing similarities between images. This method was used to find other copies of multiple images during data set formation. The images with less than 5% of differences were eliminated.

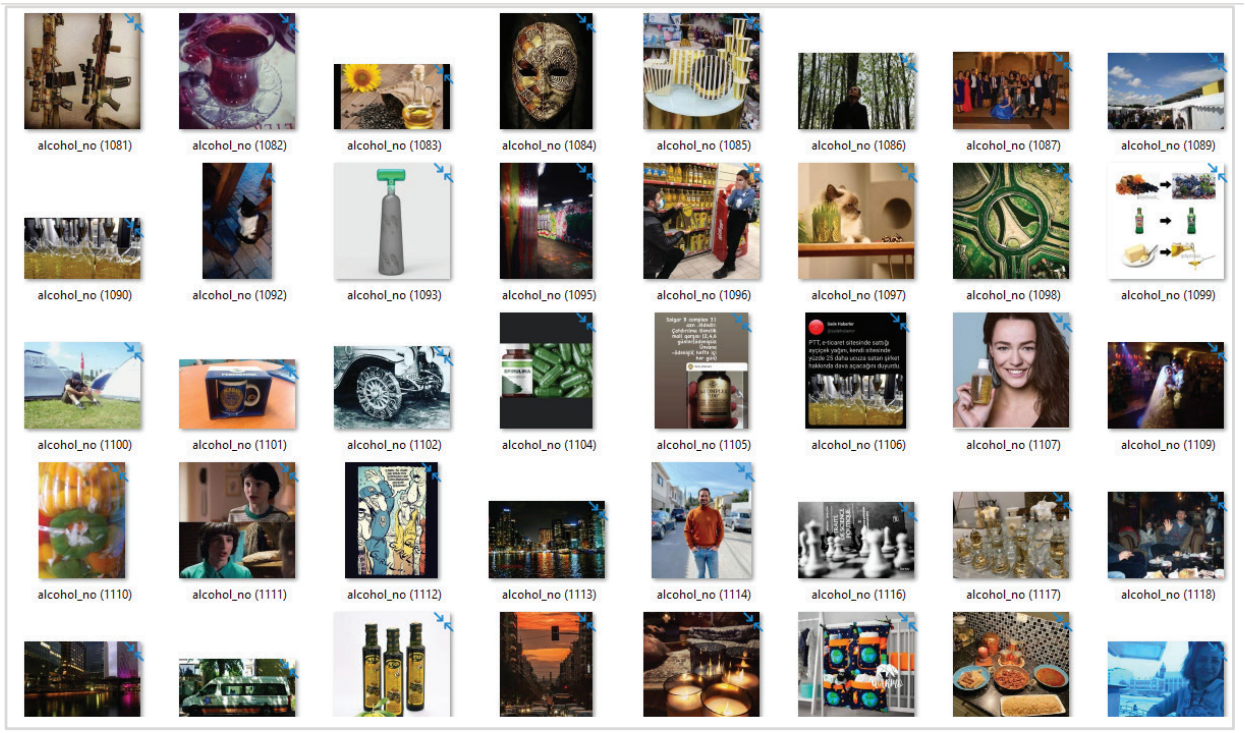
$$MSE = \sum_{n}^{i=1}(y_i - f(x_i))^2 \quad (1)\ [29]$$

Data Augmentation

Deep artificial neural networks require massive sizes of data set for the learning process. Limited amount of image data can be augmented and enhanced through various algorithms. Geometric and photometric methods are used to augment visual data [30]. Geometric augmentation is the process of creating a different image to be used in CNN learning process by changing the direction and position of the initial form of an image. These processes may be reversing, cropping, scaling, and rotating. Photometric augmentation is the process of augmentation that is applied through falsifying the chrominance channels of the image [31].

Augmentor library in python environment was used for data augmentation process. Figure 5 shows images obtained through data augmentation from an image.

Augmentor is a Python software suite that is used for data generation for machine learning problems [32]. It is possible to select probability parameters for each process. Table 1 shows the processes and parameters. Each image in



(a) alcohol_no examples

(b) alcohol_yes examples

Figure 4. Data Set Examples.

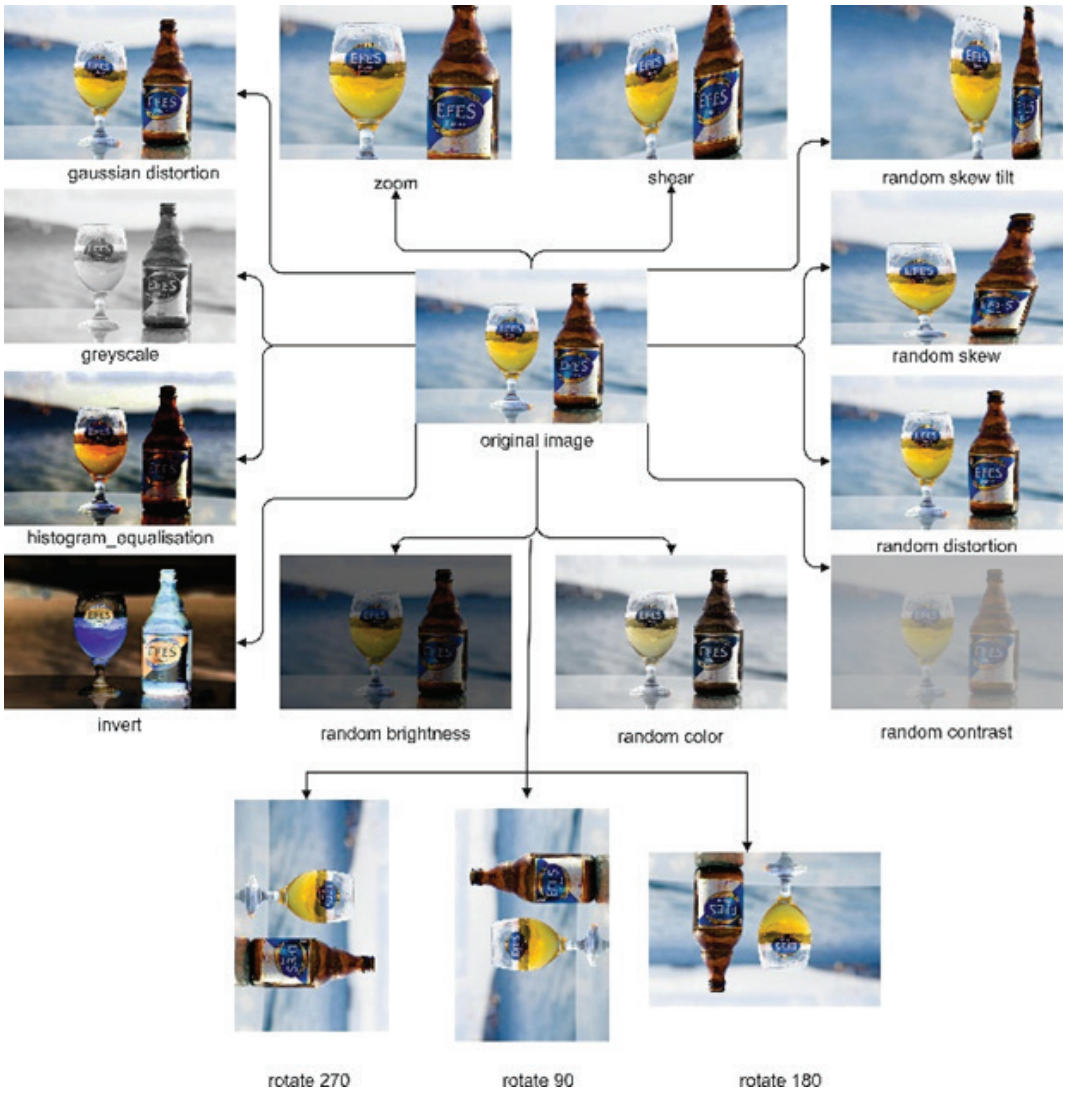


Figure 5. Data Augmentation Examples.

Table 1. The processes and parameters applied from the augmentor library

Process	Parameters
gaussian_distortion	probability=0.9, grid_width=8, grid_height=5, magnitude=5, corner="bell", method="in", mex=0.5, mey=0.5, sdx=0.05, sdy=0.05)
greyscale	probability=0.9
histogram_equalisation	probability=0.9
invert	probability=0.9
random_brightness	probability=0.9, min_factor=0.2, max_factor=0.6
random_color	probability=0.9, min_factor=0.1, max_factor=0.8
random_contrast	probability=0.9, min_factor=0.1, max_factor=0.8
random_distortion	probability=0.9, grid_width=5, grid_height=6, magnitude=4
shear	probability=0.9, max_shear_left=15, max_shear_right=15
skew	probability=0.9, magnitude=1
skew_tilt	probability=0.9, magnitude=1
zoom	probability=0.9, min_factor=1.1, max_factor=1.8

Table 2. The number of data augmentation processes and total amount of data for data sets for db_01, db_02, and db_03

	db_t	i_s	S_v	r_n	*Total Data*
db_01	279643	12	10000	3	1598572
db_02	305254	12	70000	3	4581016
db_03	311370	12	70000	3	4605480

Figure 6. Sectional ratios of data set's training, test, and validation sections.

Table 3. Data sets and data amounts of test, train, and validation

Data Set	*Total Data*	*train*	*test*	*validation*
db_00	80000	48000	16000	16000
db_01	1598572	959144	319714	319714
db_02	4581016	2748610	916203	916203
db_03	4605480	2763288	921096	921096

the data set has been multiplied using the procedures given in Table 1. In this way, the data set has been augmented.

In Equation (2), dbt represents the number of individual data, is represents the applied processes, sv represents the amount of synthetic data produced, and rn represents the number of rotations per data.

$$Total\ Data = (db_t + i_s.s_v)(1 + r_n) \quad (2)$$

Table 2. shows the evaluation of data augmentation process in Augmentor library and in terms of the amount of data produced. db_00 data set, which was used for parameter selection, was excluded from the data augmentation process.

The specified amount of sv gives the number of data randomly put into process from the individual data set per process. At the end of each process, a data as many as the amount of sv are produced. Since it was refrained from falling into redundant data repetition, the researchers tried to limit this number. 90, 180, and 270-degree rotation processes were performed separately on the data obtained at the end of the overall process.

Since CNN's were used in training, the data set was arranged accordingly. Each data set was divided into sections as train, test, and validation. Each section was divided into two as to its "alkol_var" (alcohol_yes) or "alkol_yok" (alcohol_no) status. As shown in Figure 6, the data set was arranged as 20% test, 60% train, and 20% validation.

Lightning Memory-Mapped Database (Lmdb) structure was used when creating the database. Table 3 shows the total data of these data sets, data amounts of test, train and validation.

Training Environment

Three different systems were used for training and application. At the beginning of the training process and while determining the hyper-parameters, the first system with an NVIDIA GeForce 1050Ti display card was used. In the following process, since this equipment was no longer sufficient due to increasing size of the data set, another system with an Nvidia GeForce 1080Ti display card was used. The firewall was applied on a laptop with Ubuntu 20.04 operating system, Intel Core i3 5505U processor, and Intel Graphics 5500 display card.

The training was performed by using Nvidia Digits software interface. Since the training status was visualized through Digits, the training status could be instantly tracked through visual graphics on a regular basis, and the training process could be ceased at the desired level when an undesirable result was observed.

CAFFE, which is an open-source deep learning application platform developed by Berkeley University, was preferred as the training platform. CAFFE also supports other deep learning architectures with different structures, such as image classification and segmentation [33].

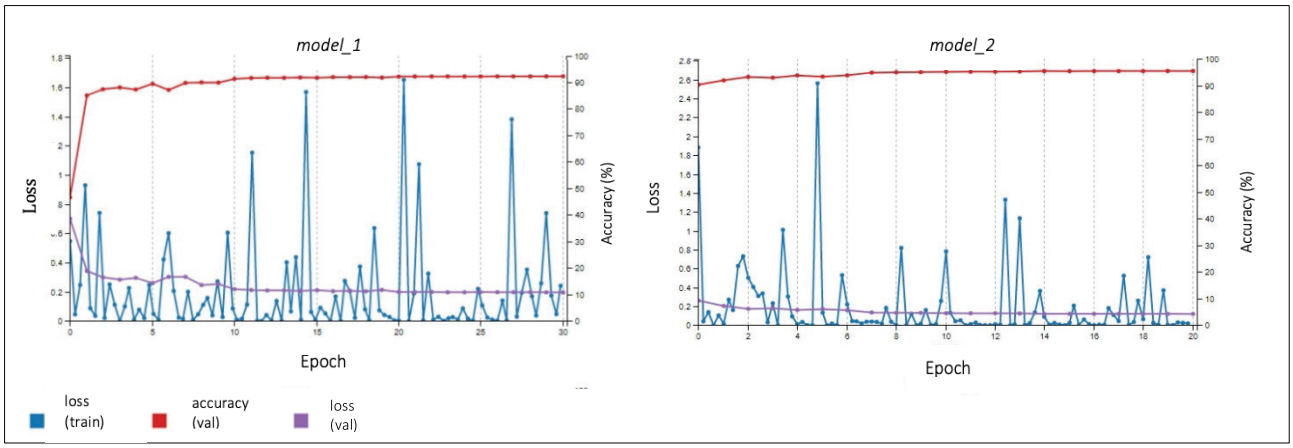


Figure 7. Model_1 and Model_2 training graphic.

Training

After determining the hyper-parameters, the training process was approached in three stages. The data set was trained for 8 days and 21 hours with db_01, model_1, 30 epoch by using NVIDIA GeForce 1050Ti display card. The data set was expanded with data, which were detected to be incorrect according to the model application test result, and db_02 was created. The model_2 was trained for 8 days and 21 hours by taking the model_1 training weights as the initial value with 20 epoch by using NVIDIA GeForce 1050Ti display card. Db_03 was created with incorrect estimate data according to the test result of model_2, and model_3 was trained in 30 epoch within 2 days and 17 hours with NVIDIA GeForce 1080Ti. Model_3 was used in firewall application.

Training graphics of model_1 and model_2 are given in Figure 7. The data sets were improved according to their error statuses, and the data set was expanded and retrained as model_3. The training graphic is given in Figure 8. At the end of 30th epoch, accuracy value was found 97.6469, loss (train) was found 0.00384845, and loss value was found 0.0697716.

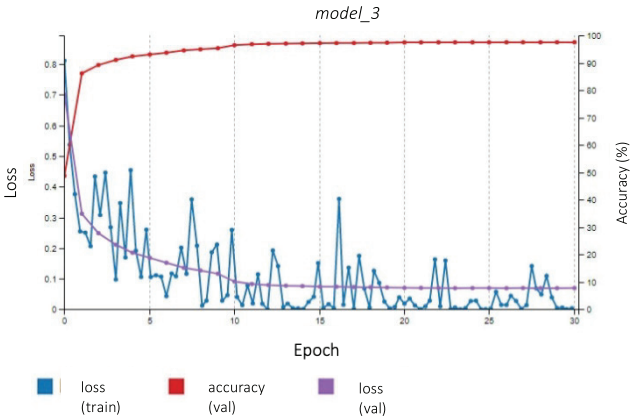


Figure 8. Model_3 training graphic

Testing of the Model

Digits interface was used in singular image tests, which is the first stage of the model. The output images of the layers, such as convolution, pooling, and normalization, can be observed separately through the digits interface. Figure 9 shows the screen shots of the results of the "alkol_var" and "alkol_yok" query on the singular image. The Images tagged (a), (b), (c), (d), (e), and (f) in Figure 9 were compared with cases including and not including alcoholic drinks in the same background. The estimation rates were found to be quite successful at the end of this comparison.

The Model's Confusion Matrix Evaluation

It is not easy to understand to the extent which the accuracy and error values obtained during model training coincide with the real values merely by reviewing the training results. Therefore, the confusion matrix rates of accurate and false classification estimation values were compared with the data that are not included within the training set. The estimation statuses of the model trained for this process were classified as true positive, false positive, true negative, and false negative as given in Figure 10.

Since there are two groups as "alcohol_yes" and "alcohol_no" in model classification, the classification values were determined in accordance with the following descriptions:

- TP (true positive): The data that include alcoholic drinks and classified as "alcohol_yes".
- FN (false negative): The data that include alcoholic drinks, however classified as "alcohol_no".
- FP (false positive): The data that do not include alcoholic drinks, however classified as "alcohol_yes".
- TN (true negative): The data that do not include alcoholic drinks and classified as "alcohol_no".

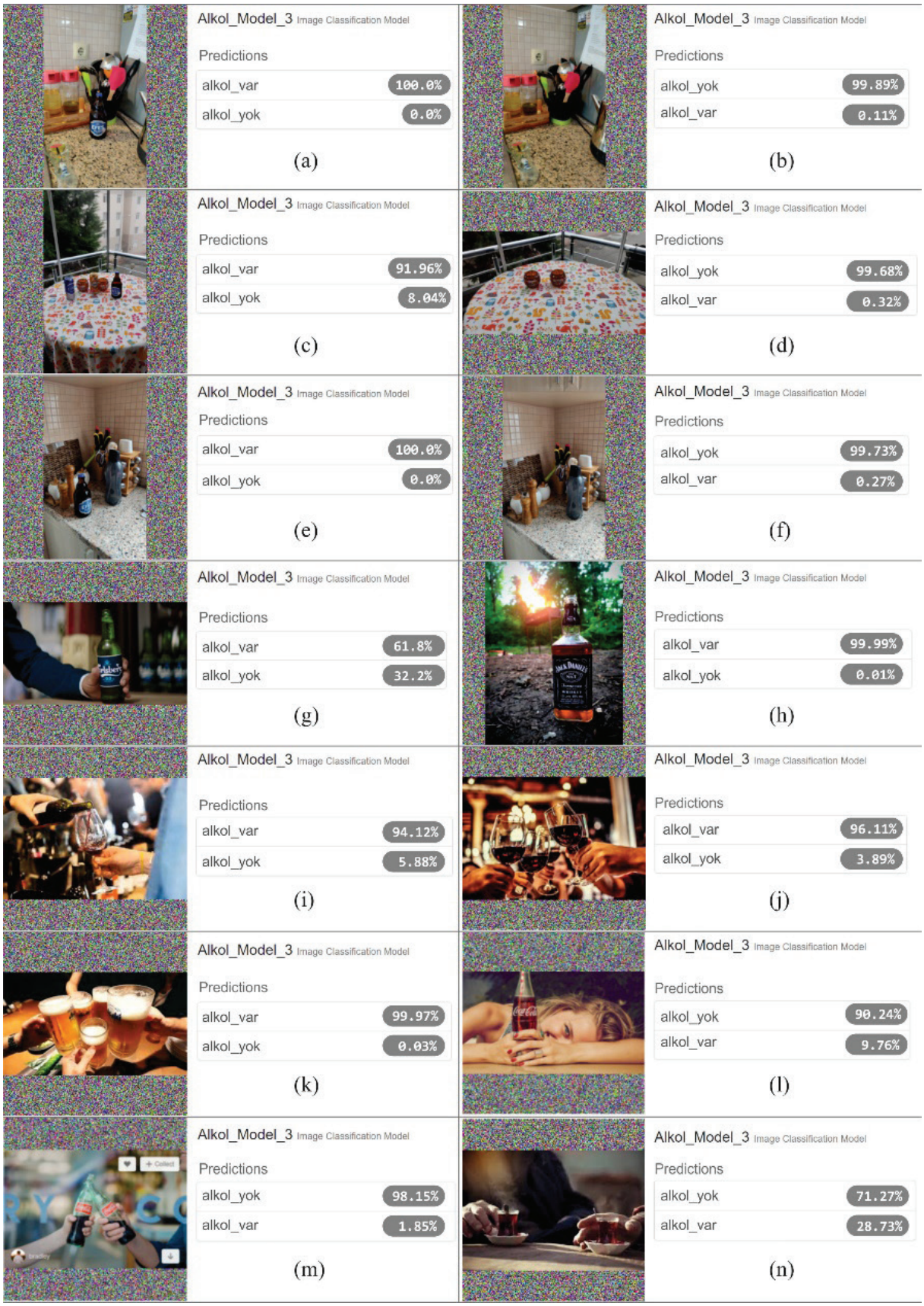


Figure 9. Examples of alcoholic drink detections through Digits interface (alkol_yok as alcohol_no & alkol_var as alcohol_yes).

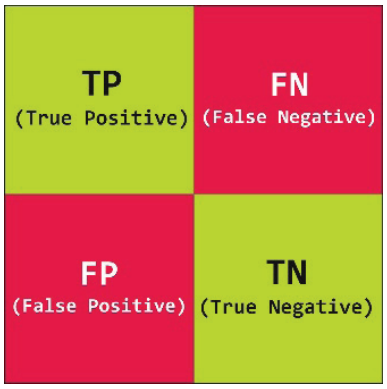


Figure 10. Confusion Matrix Evaluation metrics.

Table 4. Confusion matrix evaluation metrics formulas

Metric	Formula
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
Specificity	$\frac{TN}{TN+FP}$
F1 score	$2.\frac{\text{Precision}\,.\,\text{Recall}}{\text{Precision}+\text{Recall}}$

Figure 11. TP, FP, TN, and FN data score of seven different tests performed for confusion matrix calculation.

Table 5. Average values of precision, recall, accuracy, specificity and f1 score of seven different tests

Metrics	Average Value
Precision	0.975476359
Recall	0.797451923
Accuracy	0.940065113
Specificity	0.992612997
F1	0.877526188

In consideration of these data, the calculation metrics were evaluated with accuracy, precision, recall, specificity, and f1 score titles given in Table 4.

The test data to be used in confusion matrix calculation consist of data that were not included in training data set as training, verification, or test data. The classification process was completed by using the model trained with 154501 singular images obtained through the Internet. The images used here can actually be seen in websites in real life and they were blended for diversity. Table 4 shows the evaluation metrics for confusion matrix [34].

This test data set is performed with seven different tests and with approximately 22000 images per test. Figure 11 shows the TP, FP, TN, FN values obtained at the end these seven different tests. Table 5 shows the confusion matrix values obtained at the end of the test.

Firewall Application

The intended filtering structure is expected to detect and evaluate the images. The network traffic should be constantly checked in order to analyze whether the images contain harmful contents. Therefore, it has become necessary to create a proxy server. The entire traffic is transmitted through the proxy server; the proxy server directs the images that it detected in the network traffic towards CAFFE model file. It decides whether the image can be viewed or not according to the classification information coming from CAFFE model file. The application block scheme is given in Figure 12.

Mitmproxy tool was used for proxy server structure. Mitmproxy is a cluster of tools that provide an SSL/TLS featured observing, changing, and blocking proxy interactive for http/1, http/2, and websockets [35]. The entire flow is kept in a buffer with mitmproxy. It enables the users to process these data later, if required. The biggest advantage of mitmproxy tool is that it allows for interference with the network traffic through python command sequences [36].

In an ordinary http connection, the client makes a request to the proxy with "GET" method and the proxy conveys this to the server. In an open http connection, the proxy cannot read or impact the encrypted TLS data flow. On this connection, the proxy is nothing more than a facilitator. Thus, the information about the contents of the data flowing through this connection cannot be obtained. After the connection agreement with TLS is made through this line, all requests and responses become transparent

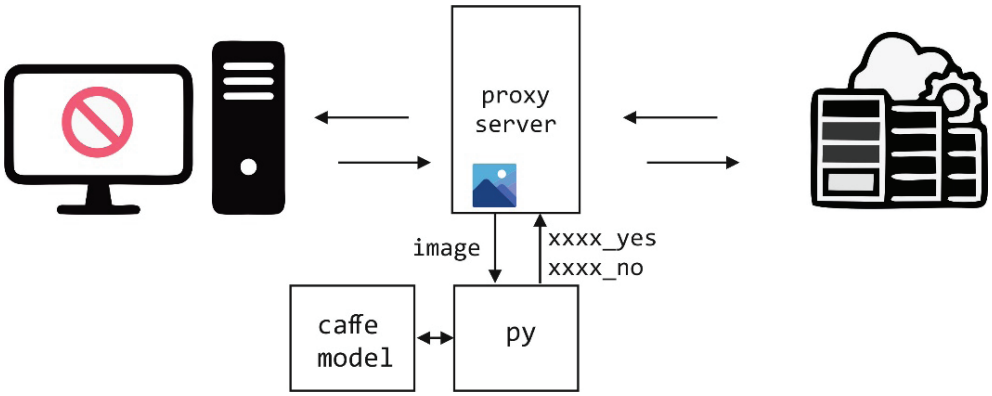


Figure 12. Application block scheme.

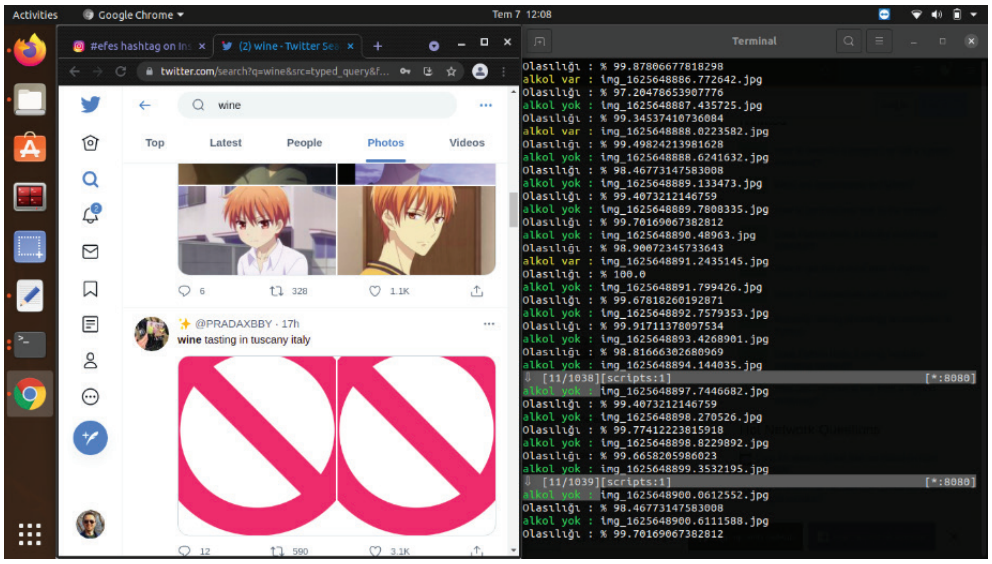


Figure 13. Screen shots of web contents with an active firewall and the terminal output.

for the proxy. Mitmproxy is located in the middle of this transparent connection flow. In decryption stage, it acts as if the server is the client, and the client is the server. For problems that may arise from certificate authority, it uses its own certificate authority and creates its own shear certificates.

First, the client establishes connection via mitmproxy and requests an "HTTP CONNECT". Secondly, the mitmproxy gives "200 Connection Established" response as if it established a connection with the client, and then it convinces the client that it is communicating with the remote server. In order to indicate the name of the main computer that in connected, it initiates a TLS connection by using SNI. Thus, the mitmproxy connects to the server and establishes a TLS connection by using the SNI main computer name specified by the client. The server gives a response to the matching certificate including CN and SAN values. After this step, mitmproxy Creates its certificate for the connection paused at the 3rd step and TLS agreement continues. The communication occurs through transmission of the requests and responses through the connection between the client and the server mediated by mitmproxy [35].

Figure 13 shows the screen shot of the integrated use of the model with mitmproxy. When the user sends a request to a website, the images sent through the network are blocked according to the estimation result of the model. In order to find out if these images are blocked, a block icon image was selected to be displayed instead of the blocked image.

Figure 14 shows the results of the screen shots of the images taken when the firewall was active and inactive according to the search results regarding alcoholic drinks performed on Twitter and Instagram.

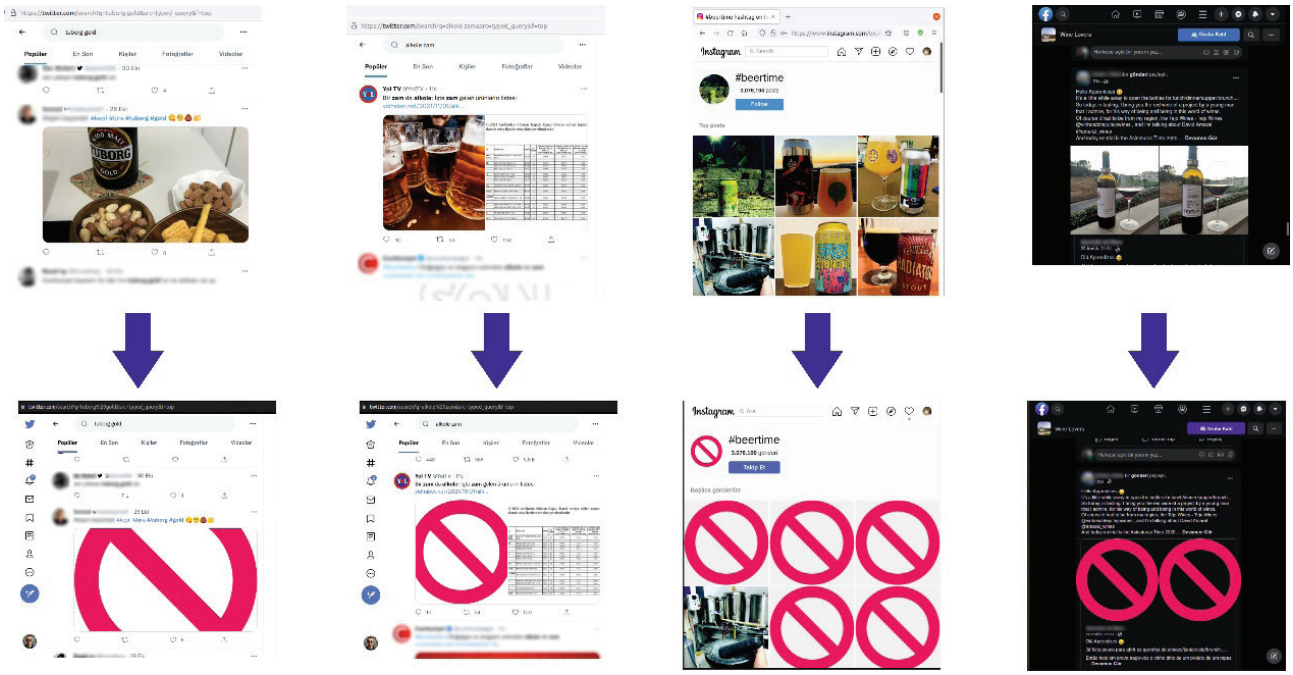


Figure 14. Screen shots of the pages viewed on the web browser when the firewall was active and inactive.

RESULTS AND DISCUSSION

A smart filtering mechanism is presented by performing image content analysis on the suggested application firewall model. The user can filter potentially harmful images by using this smart firewall. In this application model, only the alcoholic drinks were considered as harmful images, and a model was trained accordingly; the blocking process was completed successfully. However, various models can be included within the same mechanism with suitable data sets, such as weapons, violence, and 18+ adult contents. Each filtering rule can be activated or deactivated separately, when necessary.

The size of the data set, in terms of the type of the problem, is of capital importance for obtaining successful results from the training. The highest level of success was achieved only with 4.6 million image data with the means at hand. The model did not bring 100% success even in this way. However, considering the challenges of the problem, it could be regarded as a significant success. However, in order to ensure continuous success in estimations of the model, the data set may have to be updated periodically, and the training may have to be repeated with changing alcoholic or nonalcoholic drinks.

In this suggested firewall model, the image uploading speed slows down remarkably particularly because of image classification process. Graphics processor support is crucial for image classification. Setup can be performed on the proxy server through the network, however, higher process capacities for the server are recommended for efficient use. The effect of firewall on connection speed may decrease significantly with the future enhancements and developments in hardware technologies. Thus, more efficient ways for protection against harmful contents of the Internet may be developed.

Besides a firewall created through a proxy server, the trained model file can also be used in a browser add-on. In addition, a smart filtering and editing mechanism can also be used for video images to filter potentially harmful contents.

CONCLUSION

The suggested application development process was performed at six stages: collecting images for the data set, creating a data set with these images, determining and enhancing the hyper-parameters, training, success analysis through confusion matrix, and firewall application.

The present deep learning model acts as the main body for the firewall to filter the harmful contents. Alcoholic drinks were selected as the filtering subject. On the off chance of reiteration of each image selected for the data set, they were compared through means square function, and the images containing differences less than 5% were eliminated from the data set. For a successful training, the data set containing individual images have gone through duplication process, and the third data set contained a total of more than 4.6 million images.

Digits software was selected for the training. The training was performed by step-by-step enhancement with Adam optimizer algorithm by using GoogLeNet architecture in CAFFE environment. The training took 783 hours in total.

At the end of this training, 97.6469% of accuracy was achieved. It was observed that the ideal training epoch number was between 20 and 30. It was also observed that higher number of cycles had no significant impact on the accuracy result of the training.

154501 individual images were used as test data in calculation of confusion matrix. This data used for the test was performed through seven different tests with an average of 20000 images per test. These values obtained through approximately 20000 data remained constant with minor deviations. The precision was found as follows when calculated over the total values: 0.975476359, recall: 0.797451923, accuracy: 0.940065113, specificity: 0.992612997, and f1 score: 0.877526188. In coherence with the confusion matrix calculation results, the test applications also gave quite successful results.

A proxy server was required in order to control the network traffic and read the contents. Mitmproxy proxy server, which ensures flexible intervention with the network traffic with Python commands, was preferred. Thanks to mitmproxy, TLS encrypted data flow can be read, and intervened with python commands. Each image detected within the data flow is classified first through the interface created with the python commands, and CAFFE model file. At the end of this classification, an image is either blocked or not depending on whether it is harmful or not.

Tests were performed on many websites, including social media platforms, such as Facebook, Instagram, and Twitter.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

REFERENCES

[1] McCulloch WS, Pitts W. A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys 1943;5:115–133. [CrossRef]

[2] Demir I, Karaboğa HA. Modeling mathematics achievement with deep learning methods. Sigma J Eng Nat Sci 2021;39:33–40. [CrossRef]

[3] Rajkovic KM, Avramovic JM, Milic PS, Stamenkovic OS. Optimization of ultrasound-assisted base-catalyzed methanolysis of sunflower oil using response surface and artificial neural network methodologies. Chem Eng J 2013;215:82–89. [CrossRef]

[4] Zettler AH, Poisel R, Reichl I, Stadler G. Pressure Sensitive Grouting (PSG) using an artificial neural network combined with fuzzy logic. Int J Rock Mech Min Sci 1997;34:358. [CrossRef]

[5] Ma F, Sun T, Liu L, Jing H. Detection and diagnosis of chronic kidney disease using deep learning-based heterogeneous modified artificial neural network. Future Gener Comput Syst 2020;111:17–26. [CrossRef]

[6] Sabilla SI, Sarno R, Siswantoro J. Estimating gas concentration using artificial neural network for electronic nose. Procedia Comput Sci 2017;124:181–188. [CrossRef]

[7] Esen H, Esen M, Ozsolak O. Modelling and experimental performance analysis of solar-assisted ground source heat pump system. J Exp Theor Artif Intell 2017;29:1–17. [CrossRef]

[8] Esen H, Inalli M, Sengur A, Esen M. Predicting performance of a ground-source heat pump system using fuzzy weighted pre-processing-based ANFIS. Build Environ 2008;43:2178–2187. [CrossRef]

[9] Efe E, Alganci U. Determination of land cover change with multi-temporal Sentinel 2 satellite images and machine learning-based algorithms. Geomatik Derg 2023;8:27–34. [Turkish] [CrossRef]

[10] We are social. Special report - Digital 2021. Your ultimate guide to the evolving digital world. Available at: https://wearesocial.com/digital-2021. Accessed on Jul 2, 2024.

[11] Yaraş E, Yetkin Özbük RY, Çorlu P. Emmy ödüllü dizilerde alkol ve sigara ürün yerleştirme uygulamalarının içerik analizi yöntemi ile incelenmesi. Kastamonu Univ İktis İdar Bil Fak Derg 2018;20:67–84.

[12] İplikçi HG, Batu M. Digital communication and children: A content analysis of advertisements on the websites for children in Turkey. J Akdeniz İletiş 2018;29:242–256. Turkish.

[13] Uzun R. The Protection of children from media content and in media content: A study of ethical codes for children in media. J Akdeniz İletiş 2014;22:152–167.

[14] Kanbur BN. The effects of visual media and subliminal messages on child health. İstanbul Gelişim Univ Sağ Bil Derg 2020;10:94–106. [Turkish] [CrossRef]

[15] Berners-Lee CM. Cybernetics and forecasting. Nature 1968;219:202–203. [CrossRef]

[16] Lauriola I, Lavelli A, Aiolli F. An introduction to deep learning in natural language processing: Models, techniques, and tools. Neurocomput 2022;470:443–456. [CrossRef]

[17] Gupta A, Anpalagan A, Guan L, Khwaja AS. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. Array 2021;10:100057. [CrossRef]
[18] Fahad S, Ranjan A, Yadav J, Deepak A. A survey of speech emotion recognition in natural environment. Digit Signal Process 2021;110:102951. [CrossRef]
[19] Possemiers A, Lee I. Evaluating deep learned voice compression for use in video games. Expert Syst Appl 2021;181:115180. [CrossRef]
[20] Du X, Cai Y, Wang S, Zhang L. Owerview of deep learning. In proceedings of the 31st Youth Academic Annual Conference of Chinese Association of Automation; 2016 Nov 11–13; Wuhan, China. IEEE; 2016. pp. 159–64. [CrossRef]
[21] Pathak AR, Pandey M, Rautaray S. Application of deep learning for object detection. Procedia Comput Sci 2018;132:1706–1717. [CrossRef]
[22] Hu H, Pang L, Shi Z. Image matting in the perception granular deep learning. Knowl Based Syst 2016;102:51–63. [CrossRef]
[23] Tiken C. Deep learning applications. Master's thesis. Istanbul: Istanbul Univ; 2015.
[24] Dong S, Wang P, Abbas K. A survey on deep learning and its applications. Comput Sci Rev 2021;40:100379. [CrossRef]
[25] Yiğit ÖE, Alp S, Öz E. Prediction of bist price indices: A comparative study between traditional and deep learning methods. Sigma J Eng Nat Sci 2020;38:1693–1704.
[26] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE 1998;86:2278–2324. [CrossRef]
[27] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Commun ACM. 2017;60:84–90. [CrossRef]
[28] Wikipedia. List of alcoholic drinks. Available at: https://en.wikipedia.org/wiki/List_of_alcoholic_drinks. Accessed Jul 2, 2024.
[29] Sarkar N. Mean square error matrix comparison of some estimators in linear regressions with multicollinearity. Stat Probabil Lett 1996;30:133–138. [CrossRef]
[30] Taylor L, Nitschke G. Improving deep learning with generic data augmentation. In proceedings of the 2018 IEEE Symposium Series on Computational Intelligence (SSCI); 2017 Nov 18–21; Bangalore, India. IEEE; 2018. p. 1542–2547. [CrossRef]
[31] Shorten C, Khoshgoftaar TM. A survey on image data augmentation for deep learning. J Big Data 2019;6:60. [CrossRef]
[32] Augmentor. Available at: https://augmentor.readthedocs.io/en/master/. Accessed on Jul 2, 2024.
[33] Jia Y, Shelhamer E, Donahue J, Karayev S, Long J, Girshick R, et al. Convolutional architecture for fast feature embedding. In proceedings of the 22nd ACM International Conference on Multimedia; 2015 Jun 18–19; California, USA. ACM; 2015. pp. 675–8.
[34] Amidi A, Amidi S. Machine Learning tips and tricks cheatsheet. Available at: https://stanford.edu/~shervine/teaching/cs-229/cheatsheet-machine-learning-tips-and-tricks. Accessed on Jul 2, 2024.
[35] Mitmproxy Docs. Mitmproxy. Available at: https://docs.mitmproxy.org/stable/. Accessed Jul 2, 2024.
[36] Wang Y, Xu G, Liu X, Mao W, Si C, Pedrycz W, et al. Identifying vulnerabilities of SSL/TLS certificate verification in Android apps with static and dynamic analysis. J Syst Softw 2020;167:110609. [CrossRef]