Use of visual recognition to fight self-harming contents on social networks

University of Padova - School of Science

Department of Mathematics “Tullio Levi-Civita”

September 2020

Federico Brian
federico.brian@studenti.unipd.it

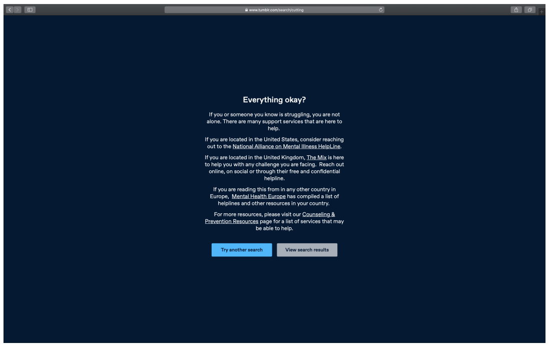
Mirko Franco
mirko.franco.1@studenti.unipd.it

Abstract

In the last years, social networks usage has seen an enormous growth. On them, not only we can find useful and positive information, but also negative material, such as visual contents representing Non-Suicidal Self-Injury[1] *behaviour. This type of visual content is proven to have a negative influence on those who see such contents, especially on young and vulnerable people, because they facilitate the normalisation and the imitation of those ones. [4]*
On social networks, and in particular on Tumblr, there is a strong and hidden presence of people performing NSSI actions who sometimes share photos representing the act of cutting. With this work, we have developed a visual recognition model able to decide whether a photo contains cuts on any body's part or not. We then have compared different convolutional architectures to find out wich one best fulfilled the task. Our final aim is to raise awareness on this delicate subject and be able to identify and support those who are hurting themselves.

1. Introduction

Internet, and social networks later, have been one of the most important revolution either for the way we search information and the way we communicate. One of the major problems is that, virtually, there is not any control on the contents we share on the net, so we can find either positive or negative material without any guarantee. It is proven that what we see online changes the way we deal with ourselves and with others [4].



www.tumblr.com/search/cutting

Everything okay?

If you or someone you know is struggling, you are not alone. There are many support services that are here to help.

If you are located in the United States, consider reaching out to the National Alliance on Mental Illness HelpLine.

If you are located in the United Kingdom, The Mix is here to help you with any challenge you are facing. Reach out online, on social or through their free and confidential helpline.

If you are reading this from in any other country in Europe, Mental Health Europe has compiled a list of helplines and other resources in your country.

For more resources, please visit our Counseling & Prevention Resources page for a list of services that may be able to help.

Figure 1. Alert for sensible contents

With this work, we propose a method to investigate the presence of visual content that represents NSSI behaviour on *Tumblr*[11], one of the most utilised microblogging platform, which has reached more than 500 milion of total blogs in September 2020[12].
Any Tumblr user can share eight different types of post: *photo*, *text*, *quote*, *audio*, *video* and *answer*, with photos dominating the distribution, representing more than 75% of the total posts[1].
Among the blogs of this social network, there is an hidden, yet strong, presence of people performing self-harming actions who sometimes share photos representing the act of cutting themselves or other related actions. Tumblr hides the research results for some specific hashtag (see figure 1), but this is not sufficient to protect vulnerable people from such contents. Moreover, it is proven that this type of content has a negative influence on young users, facilitating the normalisation of this behaviour and its imitation[4].
For these reasons, we have developed a state-of-the-art model able to recognise images representing cuts on dif-

[1] from now on: *NSSI*

ferent part of the body which could be used to study this delicate phenomena. The ultimate goal of this research is to raise awareness on this delicate subject and, perhaps, be able to identify and support those who are damaging themselves, making Tumblr, and social networks in general, a better place to explore.

2. Related Work

Self-harm is a very delicate subject, especially when everyone can access to material that either encourages or exhibits it. On the other hand, we have to keep in mind that we can not fall in censure, otherwise we invalidate the nature of Internet of be a free place where we can communicate and share ideas.

While many articles about Tumblr have been published in major press (*e.g.* when Tumblr has banned porn contents), there is not so much research so far. Chang et al.[1] studied the social network structure, giving some statistics about blogs, connections between them, posts and their sharing which on Tumblr is called *reblog*. In particular, they give a view of Tumblr as a social network, as a platform for contents generation and their propagation. They say that about half of Tumblr's visitors are under 25 years old, hence it is clear that we have to pay attention to the type of contents people have access.

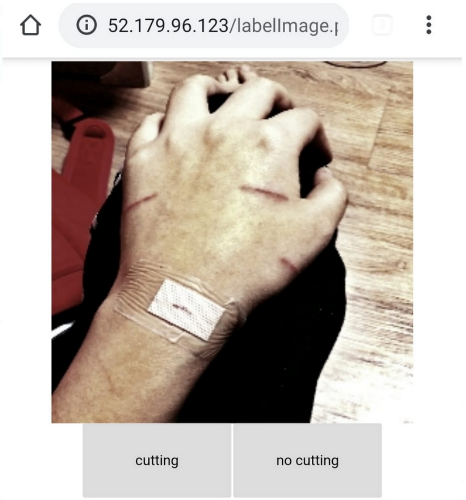
Jacob et al.[4] interviewed some people aged 16-24 recruited through Facebook ads about their's lived experience of self-harm. In particular, they investigated three distinct aspect of social media platforms for young people engaging in self-harm: the role of Internet, the influence of online imagery and the use of social media for displaying images. They say that 51.3% of young people who report self-harm have previously engaged in related internet searches for self-harm or suicide related material. Therefore, it's evident that the material we find online can encourage this kind of behaviour. On the other hand we have to consider this research is conducted with a very small sample of people. On the other hand, we have to consider this research is conducted with a very small sample of people.

Scherr et al.[6] developed an automatic image recognition algorithm for detecting self-harming contents on Instagram[8], the world's second most-used social network. They reached 87% of accuracy while classifying images as *cutting* or *non-cutting*, utilising the AlexNet[5] architecture.

3. Dataset

In order to conduct our experiments, we needed a large amount of data to train the networks with. First of all, we selected some hashtags (both in english and italian) where there could be images representing cuts, such as `#cutting`, `#selfharming`, `#autolesionismo`, `#tagli`, and the like. Then, we developed a webscraping tool capable of automatically download a significant amount of photos through *Tumblr APIs*[13] that were posted under the previous hashtags. Considering the gigantic number of images that needed to be labelled, we have built a very simple website and asked our families and closest friends to help us manually label those images (see figure 2). Avoidance to involve random users on the internet was crucial for us, because it kept away possible *trolls* that could have invalidated the entire classifying work by intentionally misclassifying the photos.

The website, as it can be seen, had a simple yet very effective structure. Users were asked to press the "cutting" but-



Cutting count: 3976

[redacted] - 1493 cutting - 8121 no cutting

[redacted] - 481 cutting - 2749 no cutting

Figure 2. The simple yet useful labelling website

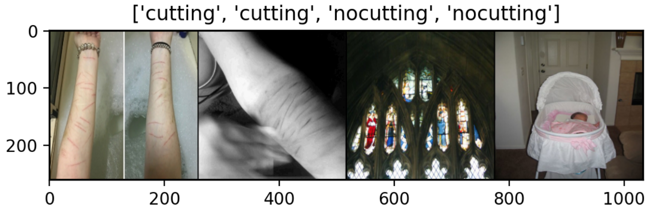


Figure 3. Extract from our dataset

ton if they recognized a self-cutting act in the depicted image, “no cutting” otherwise. In order to further improve the website users’s commitment to our work, we *gamified* this process, developing a ranking system based on the number of *cutting*-related photos found. Users could identify themselves by the IP address. Figure 2 shows the two most active “gamers”.

Once this task was fulfilled, the photos were manually controlled by the two of us, in order to ensure that they were labelled properly. They were then preprocessed in order to convert all the images to JPEG format, resize them to 256x256 pixels and delete similar (or even not human-distinguishable) images. In figure 3 you can see an extract of our dataset.

Concerning the negative instances, we used subsets of both ImageNet and “no cutting” photos, tagged as precedes by users of the labelling website. The decision about mixing two different datasets is explained as follows. The main purpose of the model is being able to recognize photos depicting the act of cutting a body part among a plethora of different - but somehow related - visual content. Therefore, if we want to develop a model as complete as possible, we also need to feed it some photo not even remotely related to the self cutting act. In such a way, it is intended to enrich the dataset, and moreover, to weaken any possibility of developing an unprecise model due to the dataset’s lack of variety. We all know how creative the internet community can be, and so we need to be prepared.

We worked with 5000 positive examples and 5000 negative ones in total, with each photo having a size of 256x256 pixels. The negatives were randomly chosen by an automated script that equally drew from both datasets - the no-cutting one and the one manually downloaded by another script available on GitHub [14].

4. Method

When we were done with the labelling task, we began to consider various convolutional neural network architectures. In particular, we chose the following four: AlexNet, VGG11 and VGG16 [7], ResNet18 [3]. As mentioned before, in [6] was used a model derived from AlexNet for a similar task as ours that obtained good performances. Thus, the reason behind this choice was pretty straightforward.

VGGNets were chosen because they are similar to - yet developed more recently than - AlexNet, having a deeper architecture and smaller filters, instead of AlexNet’s shallow architecture supporting wide layers and large filters. A graphic structural comparison between AlexNet and VGG16 can be seen in figures 4 and 5. Both AlexNet and VGGs have fully connected layer at the end, that is the only layer that we actually trained, since we performed transfer learning (fine tuning in particular). Both have achieved astonishing results in ILSVRC, with AlexNet absolute winner in 2012 and VGG16 runner-up winner in 2014. Moreover, they were trained using ImageNet dataset, which is the one adopted by us.

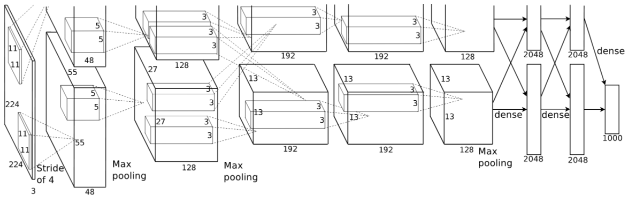


Figure 4. AlexNet architecture. Source: [5]

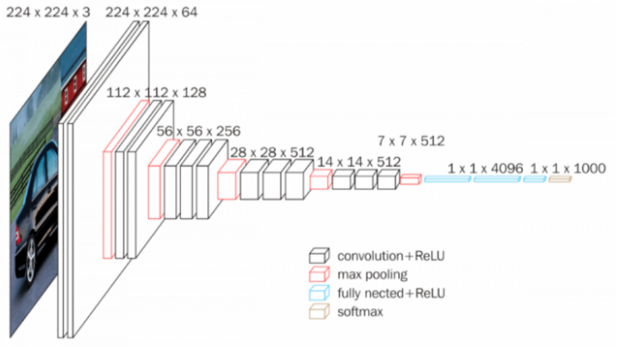


Figure 5. VGG16 architecture. Source: [9]

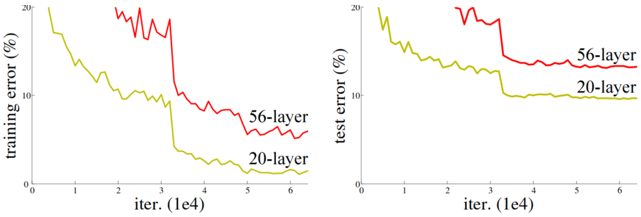


Figure 6. An example of training error (left) and test error (right) on “plain” networks. The deeper network has higher training error, and thus test error. Source: [3]

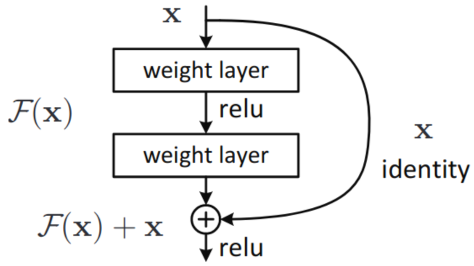

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 7. Residual learning: a building block. Source: [3]

The third architecture utilises a totally different approach than the previous two. From a theoretical point of view, a neural network with a deep architecture should be able to learn more features, gradually more abstract[2] as the deepness goes on, than a shallow neural network. The authors of [3], instead, showed that this is not achieved by simply stacking layers, in what can be called a naïve way. They proved, indeed, that over a certain limit, there are not any improvements and, moreover, the results obtained tend to be worse. This can be seen in Figure 6. Beware: stacking layers *naïvely* and/or the neural networks so built (called *plain* neural networks) are not always a bad thing. AlexNet and VGG are plain and still work great in most cases. That being said, in [3], Kaiming et al. offered a solution that is deeper and does not increase the error with respect to his shallow counterpart. First, the output from the shallow model is copied into the deeper one with the addition of identity mappings (the so-called *construction insight*). However, identity functions are not easy functions to learn. Therefore, the layers are explicitly reformulated as learning residual functions with reference to the layer inputs. The peculiar building block of which ResNets are built is shown in Figure 7.

All the experiments were developed with PyTorch[10], an open source deep learning framework, because it offers a wide set of pre-trained architectures, including the ones in our interest. Thus, we were able to operate transfer learning, which turned out to be the best choice instead of trying to induce the models to learn from scratch. In fact, the latter requires in general a greater and richer dataset with respect to the former. The dataset in our possession was indeed not sufficient to effectively train a model from scratch. In order to fix this, we contacted the authors of [6], hoping they could provide us with a part of their dataset, but unfortunately without any success.

Every architecture was loaded with the weights obtained in his pre-training. We freezed the weights' update for all the layers of the net, except for the last one, in order to have the number of output units equal to the number of classes our images was in. This is a standard procedure in transfer learning and, more specifically, in fine tuning. A PyTorch pre-trained convolutional model is usually already able to recognize basic visual components such as edges, shapes, shadows, and so on: this is generally achieved by the first layers of such a network. Thus, it is not our aim to further tune these layer's weights, because they already fulfill the tasks they are intended for. This is the main idea behind most of transfer learning techniques, such as the previous mentioned *fine tuning*. This technique allowed us to reach a global accuracy in the validation and test set that is very interesting.

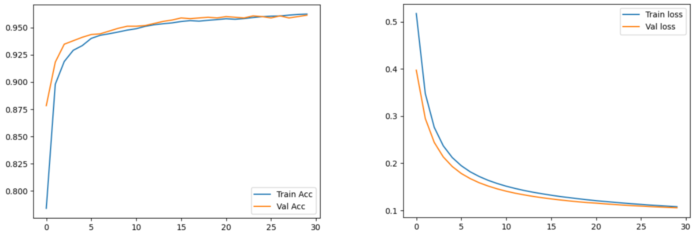


Figure 8. AlexNet training and validation accuracy/loss. Batch size: 32, epochs: 30, learning rate: 1e-5, optimizer: Adam, weight decay: 5e-2, eps:1e-6.

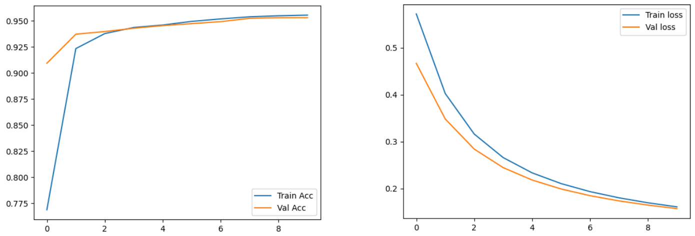


Figure 9. VGG11 training and validation accuracy/loss. Batch size: 32, epochs: 10, learning rate: 1e-5, optimizer: Adam, weight decay: 5e-2, eps:1e-6.

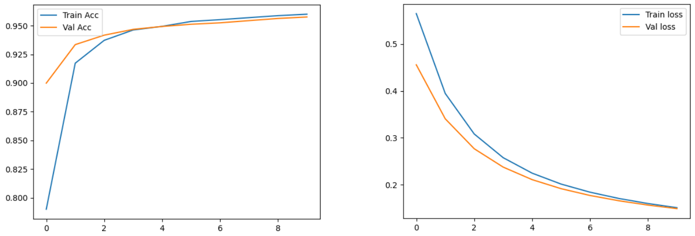


Figure 10. VGG16 training and validation accuracy/loss. Batch size: 32, epochs: 10, learning rate: 5e-5, optimizer: Adam, weight decay: 5e-2, eps:1e-6.

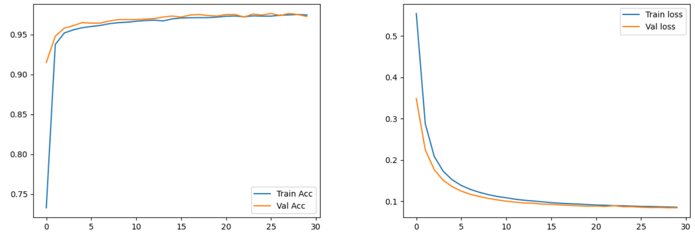


Figure 11. ResNet18 training and validation accuracy/loss. Batch size: 32, epochs: 30, learning rate: 1e-4, optimizer: Adam, weight decay: 5e-2, eps:1e-6.

5. Experiments

As mentioned before, four models with three different architectures were used: AlexNet with 8 layers, VGG with 11 and 16 layers, and ResNet with 18 layers. All of them were

[2] in other words, higher level features

trained with randomly chosen photos, both from positive and negative datasets to improve model's reliability. The negatives were also mixed using "no-cutting" labelled photos gathered with the labelling website, as well as other ones manually downloaded from the ImageNet dataset thanks to a useful tool that we found on GitHub [14]. The datasets' size of all four models was the same: 3,200 "cutting" photos for the training set, 800 for the validation set and 1,000 for the test set, for a total of 5,000 "cutting" photos. The "no-cutting" photos were as numerous as the "cutting" ones.

There will be provided a comparison between the four models developed by the authors, along with an overview of the most important hyperparameters utilised.

Some hyperparameters were the same for all four models: we are talking about the batch size (32), the optimisation function (Adam), and weight decay rate (5e-2). Adam was our first choice from the beginning, because it is a stochastic gradient descent-based optimisation algorithm that combines the benefits from both Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp). AdaGrad maintains a per-parameter learning rate that improves performance on problems with sparse gradients while RMSProp adapts these parameters' learning rates with respect to the average of recent magnitudes of the gradients for the weight [2].

Regarding the batch size and the weight decay rate, they were both found after many tries, expecially with AlexNet, and since they were a good fit for the other models too, we decided to stick with them.

In Figure 8 the results achieved by the model implementing AlexNet architecture can be seen. Initially, we began by giving it just 10 epochs, but since it kept improving its results as long as the epochs went, we decided to give it more. At the end of the 30th epoch, the model reached a validation loss equal to 0.1058 and a validation accuracy equal to 0.9613, gaining an accuracy on the test network of 96%.

In Figure 9 the results obtained by VGG11 are shown. Here, we can see that only 10 epochs are necessary to reach almost the accuracy of the AlexNet counterpart. At the end of the 10th epoch, the model reached a validation loss equal to 0.1574 and a validation accuracy equal to 0.9531, gaining an accuracy on the test network of 95%.

The Figure 10 contains the results of VGG16 model. Again, only 10 epochs were necessary to reach a very high accuracy both in validation and training set. At the end of the 10th epoch, the model reached a validation loss equal to 0.0875 and a validation accuracy equal to 0.9670, gaining an accuracy on the test network of 96%.

Finally, the last model (that can be viewed in Figure 11) implements a completely different architecture than its non-residual counterpart. As with AlexNet, we gave him more epochs because we saw that it kept performing better with respect to the epochs' passing. At the end of the 30th epoch, the model reached a validation loss equal to 0.0844 and a validation accuracy equal to 0.9728, gaining an accuracy on the test network of 97%.

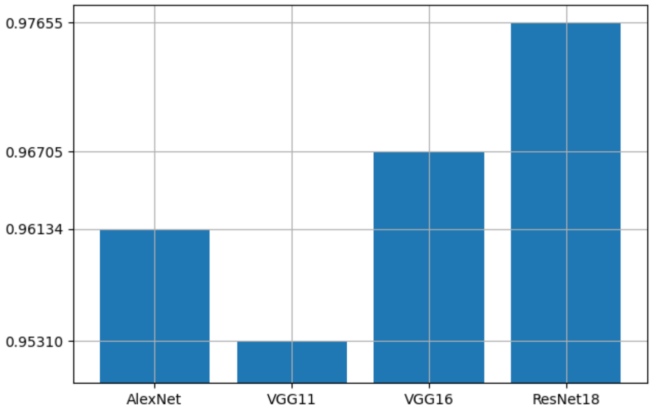


Figure 12. A comparison between the best result obtained by each of the four different models: AlexNet, VGG11, VGG16 and ResNet18.

6. Conclusion

It is now the time to gather the results obtained and to give some insights on what we have achieved. In Figure 12 the best result obtained by each of the four different models can be compared, ordered by the deepness of their architecture. It is now clear that our approach was not only the right one, but it led to some interesting results too. Perhaps not so surprisingly, there is an astonishing result obtained by the model implementing ResNet18 architecture, which not only is the deepest one (and so capable of learning higher-level features with respect to the other ones) but also has residual functions between the layers that fed input from previous layers to the current layers. This prevents the model's gradient from vanishing and speeds up learning by ablating unnecessary feature spaces. This is exactly what one generally wants, when performing transfer learning.

Regarding AlexNet, we obtained a result in the test set - 96% - that is way more better than the one claimed by Scherr et al. - 87% [6]. This could be given by the fact that they did not performed transfer learning, but instead tried to train a model from scratch. Not a single prediction about the model's results can be given when learning from scratch, because the gradient can always get stuck in a local minumum that is not so satisfying, or worse, a saddle point. Yet again - not so surprisingly - VVG nets gave very good results, with VGG11 slightly under and VGG16 slightly over AlexNet's precision, with a third of the epochs provided.

Making Internet a safer place is necessary in order to protect either young and vulnerable people. In this paper, we confronted three different convolutional architectures and proposed a way to recognise images representing self-harming contents with an accuracy of 97% on fresh images. Our model could be used either to study this delicate phenomenon or to inform the user of a possible danger in a web application. For example, it is possible to show to the user a message saying that he is sharing some problematic content.

We are conscious that this is not sufficient to detect all possible self-harming contents on social networks. They are presented in various form, not only as images. Hence, we believe that could be useful consider also text and hashtag, possibly crossing various form of data in order to better identify either potential self-harming contents and potential people who perform self-harm.

Moreover, we wish that could be possible to apply similar methods in order to limit and prevent other danger contents, such as revenge porn contents, bullying contents or those who incite anorexia.

References

[1] Yi Chang, Lei Tang, Yoshiyuki Inagaki, and Yan Liu. What is tumblr: A statistical overview and comparison. *ACM SIGKDD Explorations Newsletter*, 16, 03 2014.

[2] Jimmy Ba Diederik P. Kingma. Adam: A method for stochastic optimization. *3rd International Conference for Learning Representations, San Diego, 2015*, 12 2014.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. pages 770–778, 06 2016.

[4] Nina Jacob, Rhiannon Evans, and Jonathan Scourfield. The influence of online images on self-harm: A qualitative study of young people aged 16-24. *Journal of adolescence*, 60:140–147, 09 2017.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. Imagenet classification with deep convolutional neural networks. *Neural Information Processing Systems*, 25, 01 2012.

[6] Sebastian Scherr, Florian Arendt, Thomas Frissen, and Jose Oramas M. Detecting intentional self-harm on instagram: Development, testing, and validation of an automatic image-recognition algorithm to discover cutting-related posts. *Social Science Computer Review*, 03 2019.

[7] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014.

[8] Instagram website. https://www.instagram.com.

[9] Neurohive.io website. https://neurohive.io/en/.

[10] Pytorch website. https://pytorch.org.

[11] Tumblr website. https://www.tumblr.com.

[12] Tumblr website: about. https://www.tumblr.com/about.

[13] Tumblr website: API. https://www.tumblr.com/docs/en/api/v2.

[14] GitHub website: ImageNet Datasets Downloader. https://github.com/mf1024/imagenet-datasets-downloader.