FAKE NEWS DETECTION VIA EMBEDDED VISUAL CONTENT

MASTER THESIS

MSc. in Business Adminsitration and E-Business

SANDOR BARA

EIMANTAS URBUTIS

Supervisor: Rhagava Rao Mukkamala

Character count:159.370

15 September 2019

Acknowledgments

We would like to thank our supervisor Rhagava Rao Mukkamala for offering us his guidance and expertise throughout the thesis process, as well as, his help to find us hosting servers to develop our artifact on. A warm thank you to our families and friends for supporting us during our Master programmes. Lastly, a thank you to Copenhagen Business School for making our short time here a pleasant and memorable experience.

Copenhagen, 2019

Abstract

In recent years, given the prevalence of misinformation and fake news circulating online, in particular on Social Media, the demand for computational tools assessing the veracity of news increased. Using a Design Science approach, this thesis focused on a content-based fake news detection solution by identifying images alteration techniques that were at times used in the creation of fake news articles. In this thesis we proposed two artifacts, FaRe-PS and FaRe-GAN for the detection of two common image manipulation techniques: image splicing and image generation using General Adversarial Networks. Through the use of ensemble modeling technique and the current state-of-the-art image classification algorithm, convolutional neural network, the proposed neural networks achieved promising results. The ensemble model for the detection of image splicing, FaRe-PS was capable of detecting altered images with an accuracy of 65,05% and an AUROC score of 0,695. While, the detection of machine generated images with the devised model, FaRe-GAN, on the test images achieved an accuracy of 96,81% and an AUROC score of 0,986. To benchmark the capability of the two artifacts on the task of fake facial image detection, the judgement of the human perception was used, by means of survey evaluation. In comparison to FaRe-PS, evaluators managed to detect spliced images with a comparable accuracy of 65%. While, the developed FaRe-GAN model has far outperformed evaluators, who were deceived by generated images 43% of the time. The results demonstrated that there is a need for automated fake image detection tool and that the artifacts to a large extent effectively detect the use of two image alteration techniques used in the production of fake news articles.

Table of Contents

1.

Introduction

In today's society and in particular in the news industry the identification of fake news has become a mainstream topic of discussion since the demand for assessing the veracity of news increased given the prevalence of misinformation and false news disseminated among members of society. Misinformation circulating in the media became especially apparent in the context of politics, however vaccination and nutrition (Lazer et al., 2018) and stock information (Rapoza, 2017) have also been topics of interest to fake news publishers. This widespread propagation of fake news is especially problematic since people potentially form opinions based on alternate facts or outright lies, that can impact societies at large. If fake news is left circulating and readers become indoctrinated with misinformation, the outcome could be communities that are polarized, that are skeptical towards legitimate media organizations or even societies in which members make uninformed democratic decisions (Bakir and McStay, 2017).

"We live in an era where the flow of information and misinformation has become almost overwhelming. That is why we need to give our citizens the tools to identify fake news, improve trust online, and manage the information they receive." – Frans Timmermans, Vice-President of the European Commission

In the months leading up to the 2016 Presidential Election in the US and the 2016 EU Referendum in the UK a considerable amount of news articles have been published containing either in parts misinformation or entirely fabricated stories with ulterior motives; to affect the perception of the public (Figueira & Oliveira, 2017). Articles with headlines such as "Pope Francis Shocks World, Endorses Donald Trump for President" (Sarlin, 2018) or "Revealed: Queen backs Brexit as alleged EU bust-up with ex-Deputy PM emerges" (Dunn, 2016) turned out to be effective means of promoting political agendas or obtaining additional revenue through online traffic, by entities

that have little to no regard for ethical journalism. The former fake news article amassed close to 1 million engagements (including comments, likes and shares) on the Social Media platform, Facebook, in the three months before the US election. (Silverman, 2016). While the size of the fake news problem has yet to be adequately quantified, estimations showed that the average US citizen had been exposed to at least one or few fake news articles in the lead up to the 2016 US election (Allcott & Gentzkow, 2017).

Similar to legitimate news, fake news stories also have the potential to go viral on social media platforms, as was the case in the above example. Consequently, the focus of concern within the fake news phenomenon has moved to Social Media (Allcott & Gentzkow, 2017). We already know that a large proportion of fake news is being disseminated on these platforms (Fletcher et al., 2018) and a significant amount of Facebook's and Twitter's large userbase reads news that reaches them via said platforms (Shearer & Gottfried, 2016). The cause of concern lies within the structure of these platforms, that is significantly different from past media technologies. Whereas in the past, the distribution of news was in the hands of news organizations, think newspapers or radio as an example, in present times companies such as Google, Facebook or Twitter, have assumed control of the dissemination process; the power of who publishes what and whom the published content reaches. Taking the mantle of power when it comes to dissemination also assumes some responsibilities with veracity checking of news. However currently, content on Social Media can reach users without significant third-party filtering, fact-checking or editorial standard to be adhered to (Allcott & Gentzkow, 2017). A further concern is that users without traceable affiliation or track record of writing can at certain times reach audiences larger than reputable news organizations.

With the growing importance of Social Media in the distribution of both fake and legitimate news, policy-makers urged Social Media organizations to take actions in order to protect the public. Policy proposals were made to encourage relevant stakeholders to re-establish the tarnished trust in journalism (Newman & Fletcher, 2017) by enhancing transparency, improving media literacy and empowering readers and journalists through technology (European Commission, 2018). Likewise, scholars and researchers have shifted their focus to the problem, given that the topic of fake news rose to the prominence with no established solutions that distinguish fake news from legitimate news. Researchers devised automated solutions by detecting social bot activities (Shao et al., 2017; Bovet et al., 2017), by assessing user attitudes/stances from their replies to potentially fake news (Zhao et al., 2015; Dungs et al., 2018; Jin et al. 2016) and lastly by using factual knowledge (Ciampaglia et al., 2015), or linguistic and syntactic cues (Rosas et al., 2017) within the content of fake news articles. Similarly, the purpose of this thesis will be to develop an automated

content-based detection solution, however focusing on a yet unexplored element in fake news articles, that is images embedded within these articles.

Visual cues are already important manipulation tools within fake news to provoke emotions, often negative, from readers (Shu, 2017). It is expected that with the rapid advancement of technology and new image manipulation techniques surfacing in the future, fake news articles will become even more deceptive. Thus, visual content detection tools will be essential in the quest to reduce misinformation circulating online and to re-establish the trust of the public in the news media.

1.1 Motivation

We might live in a world where the information presented to us, when we read a book, browse online or sit in front of a television, contain misinformation or outright propaganda, however up until now we as humans could trust our judgment and our perception of reality. Once we live in a world where videos, images or audio cannot be trusted, we will awake to the reality that anything can be faked. While this notion might sound dystopian and a state that is far from materializing, there are already warning signs pointing towards it.

Last year, several videos surfaced on the Internet, known as “deep fakes”, in which faces or voices of influential people were superimposed on other videos, skewing the image of said people by putting words in their mouth. As examples, Donald Trump urging the people of Belgium to withdraw from the Paris climate agreement (Schwartz, 2018) or Mark Zuckerberg talking about being in control of billions of people´s stolen data (O’Neil, 2019). These videos were constructed using a machine learning technique called General Adversarial Networks (GAN), that synthesizes new data based on existing data. It is not far-fetched to believe that said technology could bring additional opportunities to fake news creators; to disseminate misinformation, manipulate crowd opinion and push already fringe groups towards their pre-existing beliefs.

Image manipulation utilizing splicing, the act of creating a single image from multiple images, is yet another problem in the news industry. One might say that it is even more prevalent than “deep fakes”, given that it is a relatively easy technique to use both in terms of expertise and technology required. The results of image splicing can be convincing enough to fool even the trained eyes of news editors. In some cases, it was a widely circulated manipulated image on the Internet that was taken by reputable media outlets. The Sikh man, whose image was modified to include a bomb vest and his iPad switched to a Quran, suggested as one of the perpetrators of the Paris attack in

2016 on Twitter later taken over by one of the largest Spanish newspapers (Warren, 2016). Or another case where an image manipulated by the Iranian state media, that tried to mask a failed Iranian missile test, ran on the front page of prominent US newspapers (Nizza and Lyons, 2008). There are many other examples of image manipulation by splicing, as it is a relatively easy technique to add "proof" or to eliminate readers' suspicion of news falsification.

We, as avid readers of news, believe that journalism has had a very well-defined place in society; to bridge the gap of information asymmetry between events and readers who want to be aware of said events. However, in bridging the gap journalists have also been responsible for checking the veracity of information, that otherwise might be a costly task for individuals. With the rise of new technologies, as in algorithmic news dissemination on Social Media and "proof" creation techniques, the role of fact-checking has partially shifted. Individuals will have to either continue placing their trust in news organizations at a low "cost" or resort to trusting their judgments at a higher "cost", that is the effort needed to assess the legitimacy of news. In both scenarios, readers and news distributors/creators alike can benefit from technology that not only simplifies but often improves the process of news authentication. Our motivation is to put our passion in Machine Learning to good use by exploring a solution that can contribute to the detection of fake news and the re-establishment of trust in visual content within news.

1.2 Problem formulation and research questions

In accordance with the previous paragraphs and the identified research gap in content-based fake news detection, this thesis will investigate how machine learning can be used to detect manipulated imagery and how such solution could be leveraged in detecting fake news content. While this type of news manipulation is still in its infancy, it is expected to grow at an increasing rate parallel to that of technological advancement. First and foremost, this thesis will contribute to existing knowledge by developing a machine learning solution for fake image detection, and second it will provide a proposal for implementing said solution into the existing news distribution and consumption ecosystem. To this extent the following question will be answered:

1. How can machine learning be used to detect fake visual content embedded within fake news?

To further guide the research, the following sub-questions were created that assess the viability of candidate machine learning solutions for fake image detection and the effectiveness of the final proposed solution in comparison to human perception:

a. To what extent can a neural network solution solve the detection of common image manipulation techniques?
b. How effective is machine learning in detecting fake facial images in comparison to humans?

The results of the research will contribute towards existing research on fake image detection using machine learning. Furthermore, towards fake news detection as one of the first proposals for detection implementation using manipulated images.

1.3 Scope and Delimitation

Although there are numerous ways images can be manipulated and to an extent, most images within articles receive some minor treatment, this research is solely concerned with the detection of manipulation techniques that significantly alter the meaning or message of an image. Images embedded in fake news articles often serve the purpose of either provoking emotions within readers or serve as "proof" to sway the beliefs of readers. Thus, to achieve said purposes through fake images, the meaning or message of the image must be modified. Through the detection of these significant alterations in images, one could identify fake images that are present and consequently identify fake news articles. To this extent, the detection of two common image manipulation techniques was investigated: image splicing and image generation using GAN.

Likewise, there can be a multitude of ways in which these alterations can be detected using machine learning. However, some machine learning algorithms require a considerable amount of domain knowledge within image forgery or extensive feature engineering to find the right representation of data. To alleviate these shortcomings and to limit the scope of this research, the machine learning algorithm investigated as part of this research was Convolutional Neural Networks (CNN). CNN was deemed to be a suitable machine learning method for the problem given that it needs little preprocessing and automatically represents features, and it is the current

state-of-the-art machine learning method when it comes to image recognition, object detection, etc.

Towards creating a machine learning solution that can identify these image manipulation techniques, a large number of image examples were needed that were created using the two techniques. Until the time of identifying the research problem, there was no comprehensive dataset gathered of fake imagery in news articles in the magnitude needed to train a machine learning model on. Thus, alternate non-news related image datasets were found that were produced using the manipulation techniques mentioned above. These image datasets contained facial images, both legitimate and altered using splicing or generation. Ideally, the images within the dataset should have been diverse enough (in what is presented on the image) to account for any type of imagery contained within fake news articles. However, such enormous dataset did not exist before this research, and the training of such machine learning algorithm would require a considerable amount of computational power, consequently making it technically infeasible within the scope of this thesis.

1.4 Thesis structure

In order to answer the research questions this thesis is structured as follows: In the next chapter, fake news will be defined and concepts revolving around the phenomenon of fake news, that will serve as a discussion point on how the developed machine learning solution can be put to practice. In chapter 3, machine learning and neural networks will be introduced to help the reader understand the methodology behind the fake image detection solution. Chapter 4 explains the methodology of collecting training data, the processing pipeline of data, finding candidate neural networks models and how the machine learning solution will be evaluated, including a comparison sample taken from human evaluators. In chapter 5 and 6, the results of the final neural network will be presented, followed by a discussion on the effectiveness and shortcomings of the solution and lastly a proposal for implementation in the news dissemination process. Finally, the limiting factors will be discussed that point towards suggested paths to extend the research and Chapter 9 that offers concluding remarks to the thesis.

2.

Theory and related work

The recent high-profile political events of the Brexit Referendum and the U.S presidential election of 2016 sparked a significant amount of interest in the phenomenon of "fake news", which is believed by many to have played a significant role in shaping the outcome of both events. While the expression itself since then has entered into the mainstream vocabulary, often being used in different contexts and with varying understanding of what "fake news" entails, the present paper needs to put forth a uniform definition that will be used throughout the rest of the thesis.

Lilleker, a researcher in the field of political communication sets forth the definition: "fake news is deliberate spread of misinformation, be it via traditional news media or through social media" (Lilleker, 2017, p. 2). His definition entails the requirement for a news piece to be categorized as fake news: (a) to be deliberate, meaning that it was created in bad faith and with an ulterior motive and (b) carrying misinformation, meaning information that is fabricated and not factually correct. While these two concepts are recurrent in the definition of other researchers (Lazer et al., 2018; Gelfert, 2018; Tandoc, Lim & Ling, 2017a), the definition can be extended to include the level of truthfulness of the information contained in fake news pieces. Bakir & McStay (2017) argue that not all fake news pieces rely entirely on misinformation. Certain fake news pieces carry some amount of facts that are incorporated with misleading elements either in context or in content. The misleading elements are often shocking or outrageous with the intention of creating sensationalist news. (Gelfert, 2018; Tandoc, Lim & Ling, 2017a) Lastly, a recurring requirement in some of the definitions was the appearance of fake news. Fake news mimics the format and the appearance of traditional news pieces created by reputable news organizations, thereby attaching credibility to the source of the fake news. (Levy, 2017; Lazer et al., 2018)

Building upon the elements mentioned above, for news content to be regarded fake, it should meet the following requirements:

(a) Contains fabricated or misleading information either fully or in part (in conjunction with its context)
(b) Written deliberately with the intention of instilling falsehoods in its target audience
(c) With a format that resembles that of other credible sources within the news industry.

2.1 News industry

The rise of the Internet proved to be a significant vehicle in bringing change to the news industry. Previously the industry standard was a linear business model in news publishing, where news organizations used a single channel to distribute news articles to readers, either in print, radio or television. through

Fletcher, Graves & Nielsen (2018) researched news reading habits on a sample of 40,000 people world-wide revealing that a single channel strategy indeed cannot mount to success for news organizations in the 21st century. The study revealed that only 32% of consumers access the site of a newspaper directly, whereas search, social media, etc. related new seeking activities (in other words mediated) amount to 65% consumers. News organization increasingly lose control on two essential facets of the news publishing: topic selection and distribution of news articles.

Search engines and social media removed the previously complete control of news organizations over the distribution of their articles (Martens et al., 2018). Search engines work like marketplaces where the needs of the consumers are matched with content producers. However, in search queries rank matters, therefore getting the top placement in queries is a must for publishers. Frequently read articles reach the top and therefore get even more readers, while bottom ranking articles that are rarely read drop even more down as they are barely visible to readers. Social media sites such as Facebook or Twitter work very similarly, they create a tailored content feed including news that is relevant to a given person based on said person´s interests and friends´ interactions (Allcott & Gentzkow, 2017). Said content tailoring is argued by researchers to create opinion polarization and echo chambers (Barbera et al., 2015), which will be discussed in more detail later on.

Topic selection is another significant facet of news publishing that organizations started to lose control of. Lee & Tandoc (2017) argued that the loss of control is partially due to audience

feedback that is: (a) instantaneously recorded and reported (b) from a wider audience than before and (c) more comprehensive in the form of textual (comments, opinion posts) and numeric (web analytics) feedback. In a similar vein, Welbers et al. (2016) found that said audience feedback directly manifests in the choice of topics covered by news organizations, saying that: "topics that have attracted many clicks in the past tend to be covered more often". Analogous conclusions have been found in the relationship between audience interest and news coverage. Studies have concluded that audience interest in specific topics, measured in search query volumes (Ragas et al., 2013) or discussions in online forums (Lee & Tandoc, 2017) positively influence the coverage of these topics. Meaning that the traditional roles of agenda-setting have shifted; journalists act upon the interests of the audience and not the other way around. Likewise, not only topic selection is influenced by audience feedback, but to some degree the content itself. The use of some form of visual content in news articles by journalists can be attributed to audience feedback as verified by Tandoc (2014). Articles with photos and videos garner a lot more clicks than articles without them.

Catering to audience feedback, writing about what readers want to read rather than what they need, devalues the role of journalism in a society, that is informing the public on the events happening in the outside world (Lee & Tandoc, 2017). Furthermore, relying on feedback from audience also promotes ideological isolation, as readers tend to read articles that are within their spectrum to confirm their beliefs (Flaxman et al., 2016). As such news organizations that put much emphasis on the wants of the audience create an information imbalance, where readers are only exposed to one side of an ideological spectrum. A recent example of catering to audience feedback gone wrong is Facebook´s replacement of trending topics editors with an algorithm relying on web analytics. The algorithm without human supervision started ranking fake news stories at the top of trending topics (Thielman, 2016). These fake news stories are likely to achieve virality, however, at the expense of information quality. Relying on merely audience feedback as witnessed in the situation of Facebook´s algorithm is a failure in the making when it comes to filtering out non-factual information.

2.2 People and news consumption

Alike many other industries technological advancement, in particular the rise of the Internet and social media platforms, changed both how the news industry publishes news and how people consume news. Social media platforms, which at the start served the purpose of connecting people

and foster communication in real-time, became increasingly intertwined with the news industry, to the point where these platforms became one of the go-to sources of news for people (Rochlin, 2017). A recent study conducted by Pew Research Center estimated that 62% of U.S adults occasionally get news from Social Media and 18% of them do so often. Not surprising considering that more than 2/3rd of both Facebook´s and Twitter´s users also get news from these platforms, according to the survey (Shearer & Gottfried, 2016). According to Gild de Zuniga et al. (2017) Social Media potentially makes users feel that there is an abundance of news reaching them via their news feed and social media, consequently decreasing their need to seek out news sources themselves directly. This user perception is also supported by Allcott & Gentzkow (2017): "in these circumstances, users see themselves in an environment of ambient news, where you could be led to believe that you are staying informed sufficiently".

However, the current news industry climate also holds challenges that are yet to be addressed. In a recent survey carried out on 18,000 people across nine countries by Reuters Institute, it was found that there is a sizeable minority that does not trust either traditional media (25%) or social media (41%) to help readers distinguish fact from fiction. Furthermore, in the case of traditional media, 67% of the distrusting consumers highlighted some sort of bias as the primary reason for the lack of trust. Whereas in the case social media the lack of trust stemmed from unreliability (35%), no checks on authenticity (25%) and news being agenda-driven (24%) (Newman & Fletcher, 2017). Thus, the previously held role of intermediaries acting also as gatekeepers, in the sense of authenticating information and granting credibility to it, has dropped. Metzger et al. (2010) observed that the role of gatekeepers has started to shift to one´s peers; people rely on peer-to-peer credibility assessment. This notion is especially probable considering that web-based tools enabled new social processes for assessing information credibility and trust, e.g. review sites, discussion forums. Although individuals turn to external means of credibility assessment, it is often not the first step in the authentication process. At the most basic level misinformation is determined by the reader's judgment, meaning cues in the way information is presented as well as its origins. Only if said approach turns out to be unsuccessful in determining the reliability of an article, individuals turn to external forms of authentication, e.g. rely on their peers (Tandoc et al., 2017b). As such in the current climate of the news industry both professional journalists, Social Media companies and friends/followers act as information gatekeepers who vet the authenticity and relevance of news content.

Peers and friends as gatekeepers are even more prevalent when it comes to Social Media. Turcotte et al. (2015) investigated opinion leaders, who are knowledgeable and trusted sources of information, influence on opinion follower´s trust in news articles and news organization. The

hypothesis was that a recommendation to a news article by a trusted friend would increase trust in the information contained in comparison to directly accessing the news article. The findings indicated that news shared by a friend on Facebook "is perceived as more trustworthy than stories received directly from the media outlet" (Turcotte et al., 2015, p. 529); likewise, it also increased trust in the news organization itself. Although users do not trust Facebook itself as a reliable means of news authentication according to the survey by Reuters Institute, they find trustworthy sources in their friends and peers on said platform.

While trusting peers can be a reliable heuristic in filtering out news articles carrying misinformation, only particular peers are judged to be trustworthy. Spohr (2017) suggests that consumers are prone to discredit attitude-challenging information, by either avoiding attitude-inconsistent information or counter arguing it, a phenomenon called selective exposure. With that in mind, people seek information that is partial to their pre-existing beliefs and experience, while they might disregard others and the person carrying said information. Bakshy et al. (2015) conducted a study on 10.1 million Facebook users in the US and have concluded that in comparison to algorithmic personalization on Facebook, users´ choices were more impactful in limiting their exposure to attitude-challenging information. In effect selective exposure can lead to homogeneous groups of people existing in a vacuum, where discussions are self-reinforcing rather than thought challenging. Spohr (2017) described these homogenous groups as feedback loops or echo chambers, which were confirmed in Facebook groups by studies such as Jacobson et al. (2016).

2.3 The role of Social Media in fake news distribution

In the previous paragraphs, it was established that Social Media platforms increasingly became the cornerstones of news consumption, consequently also of fake news distribution. Guess et al. (2018) in their study on news consumption before the 2016 US election estimated that 27,4% of Americans older than 18 had read at least one article on a pro-Trump or pro-Clinton fake news site. Interestingly, Facebook and Twitter were among the top-4 websites that these readers have visited before the fake news website. Their study is one of the first studies finding quantitative evidence that verify Social Media channels, in particular Facebook, as a critical element in the distribution of fake news. Later studies, using other means of measurement, have also confirmed this notion of fake news sites actively using Social Media to promote their messages. Looking at

the interactions data (shares, comments, reactions) on Facebook in Italy and France, Fletcher et al. (2018) were able to measure a comparable average user interaction between real and fake news. In particular in France, where a small number of fake news sites received almost as many interactions as established news providers. As an example, a right-wing blog called La Gauche m'a Tuer has received an average of 1.5 million monthly interactions in comparison to Le Figaro's and Le Monde's 1.9 and 1.7 million interactions, two of the most established newspapers in France. Similarly to France, in Italy a few false news outlets outperformed more established news organizations such as RaiNews, the Italian public broadcaster.

A few researchers have attempted to identify the reasons behind fake news articles performing well on Social Media. As mentioned earlier, fake news likely contains misleading elements that are either shocking or outrageous with the intention of creating sensationalist news. Vosoughi et al. (2018) confirmed this notion using Twitter data between 2006 to 2017 to conclude that fake news is more novel than real news. Furthermore, that fake news being novel are more likely to be shared by users on Twitter, and the diffusion of said news goes "significantly farther, faster and deeper" (Vosughi et al. 2018, p. 2). Whereas novelty might not be the only reason why users engage with fake news and voluntarily share them, it can be assumed that novel information is more interesting to users and consequently more worthy to share on Social Media platforms.

Another reason why Social Media could be viewed as a great conductor of fake news online is the way opinion/article sharing works on these platforms. As an example, any user can share "news" online with a single click to their immediate circle, without any sort of editorial standard or verification of information veracity (Allcott & Gentzkow, 2017). Social Media platforms although act as distributors of news, they do not want to assume the role of media companies in terms of vetting the authenticity of information. (Martens et al., 2018) Social Media companies would rather abstain from limiting free speech, only taking actions against hate speech and violence, thus giving leeway for fake news to spread on their platforms. (Lazer et al., 2018) Ultimately, these actions leave the fact-checking part to either the readers themselves or to the peers of readers.

The third reason for fake news spread on Social Media is the prevalence of social bots. Through activities such as mass liking, sharing and searching information, social bots spread fake news across users and garner artificial interest in fake news articles (Lazer et al., 2018). In a white paper released by Facebook in 2017, it was estimated that around 60 million bots performed these activities on their platform during the 2016 US Elections. Shao et al. (2017) analyzed the activities of these bots on Twitter and concluded that the most successful sources of political fake news were supported by networks of social bots. Social bots also employed the influence of popular users on

Twitter to not only spread but to give a sense of authenticity to fake news. Said strategy proved to be particularly effective given that it falls in line with the trust dynamic between opinion leaders and followers described above.

2.4 Fake news prevention

As a response to the growing popularity of fake news online, especially before the 2016 US Election and the British Referendum in 2016, prompted entities, policymakers and Social Media organizations, to take initiatives that prevent the phenomenon that is fake news. As an example, Facebook in 2018 proposed changes to both its algorithm and its news feed function. These changes were aimed to tackle social bots that boosted the engagement to fake news articles, create transparency when it comes to source of content and to slow down the distribution of news articles by reducing news feed to content coming from immediate friends.

Likewise, policymakers such as the High-Level Group on fake news, established by the European Commission, set forth a policy proposal based on enhancing transparency, improving media literacy and empowering readers and journalists by means of technology. Said proposal, in the short-term, suggested a self-regulatory approach, by including stakeholders in the industry to tackle the problem of fake news. (European Commission, 2018). As a main element of the proposal, the High-Level Group on fake news called for the creation of a Code of Practices, that defines the roles and responsibilities of stakeholders. In particular social networks, that according to the High-Level Group can: “impact public opinion by sorting, selecting and ranking news and information via their algorithms” (European Comission, 2018; p. 32); therefore are detrimental towards building a news landscape in which both publisher and intermediaries are trusted, and end-users are able to make informed decisions. However, said proposal is yet to be put in action (European Commission, 2018), consequently its viability cannot be assessed.

Parallel to policymakers and social media organizations, independent researchers also devised technological solutions that can detect news articles containing misinformation. These solutions are to a large extent automated and are based on one of three characteristics: the *content* of an article, the *source* of the article or the ones promoting it and the *reader response* to said article.

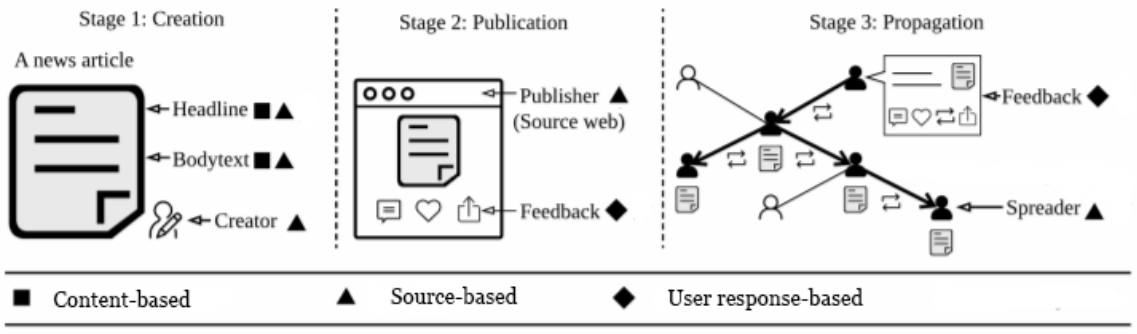


***Figure 2 -1**: Possible directions for the detection of fake news*

The *content* of an article can be assessed for containing misinformation in a multitude of ways. Linguistic, semantic and syntactic cues among others have been proposed as potential solutions for detecting deceptive news articles. To some extent, even automated fact-checking can provide a beneficial solution to the problem as demonstrated by Ciampaglia et al. (2015). The proposed solution leveraged large-scale information– repository, e.g. Wikipedia, that comprise of factual information gathered by humans, to dissect and compare claims made that are potentially consisting misinformation. To this extent, Wikipedia "infoboxes" were used as knowledge graphs coupled with network analysis to compare the truth to the new statements. Attempts to use linguistic and syntactic cues have yielded even more promising results. Rosas et al. (2017) assessed the differences between real and fake news using punctuation (e.g. dashes, commas), readability (e.g. number of words, complex or long words) and psycholinguistic features (e.g. words carrying perceptual, emotional connotations) across a fake news dataset in six different domains and a celebrity news dataset. Their findings indicate that fake content uses significantly more positive word, more punctuation characters, more words to describe present and future and lastly more words that are perceptive: "hear, see, feeling and positive emotions categories" (Rosas et al., 2017; p. 8). These results are congruent with the stylistic differences identified during the definition of fake news. In that fake news are aimed towards deceiving users not only by incorporating misinformation but by exaggerating expression and invoking strong emotions. Thus, to an acceptable level (74% accuracy on the fake news and 73% on the celebrity dataset) fake news and legitimate content can be differentiated using a classifier comprising of psycholinguistic, readability and punctuation features.

Fake news articles can also be identified during the distribution process, who is the original *source* and who are the entities disseminating said articles to newsreaders. In the previous paragraphs, it was asserted that social bots are to a large extent part of the fake news dissemination process (Shao et al., 2017). Social bots can be detected by tracing the connection network of users and

Social media use patterns. Bovet et al. (2017) through the analysis of diffusion networks on Twitter, found that social bots retweeted more people and are more retweeted on average, while still being unverified or unknown users or nonpublic figures of Twitter. In contrast, diffusers of legitimate news were found to be in the majority verified accounts belonging to news outlets, journalists or people of influence. Furthermore, these bots existed in a homogenous bubble, meaning that they were interacting with the same cluster of users and were engaging in collective behavior. Some examples of collective behavior were assessed by Cai et al. (2017) finding evidence that bot activities were unlike human behavior; social bots were heavily engaging in Twitter activity during weekdays and during the same time span each day.

A substantial body of work has been devoted to detecting fake news articles through *user/reader responses*, where the stance and opinion of users are used as indications to the credibility of a news article. Zhao et al. (2015) created an early detection system that finds reply posts of users that are expressing skepticism or raise the question about the veracity of the information in the original post. The key finding of the paper was that throughout the fake news diffusion process there are a few posts that raise these questions early. Thereby using these questions as detectors one can find post clusters that might feature fake news articles within. In similar vein, Jin et al. (2016) proposed a detection tool leveraging conflicting viewpoints from tweets related news using a topic modeling method. The underlying assumption of the research was that truthfulness of news could be assessed using an average credibility score of tweets made, both opposing and supporting, in relation to the original content. Lastly, Dungs et al. (2018) employed the stance of user responses to detect intentionally falsified information. In their detection system user responses to a particular post, referred to as crowd stance, were categorized according stance, whether the content of the user response was supporting, denying, questioning or commenting on the original post; achieving an accuracy of 80% across 173 Twitter threads containing misinformation.

3.

Background

The following chapter serves as an introduction to the field of machine learning and concepts within, in order to understand the methodology behind the derived solution to the fake image detection problem. To this extent, the theories behind deep learning and neural network will described, with an explanation and justification for the chosen algorithm of this thesis, Convolutional Neural Networks (CNN). Further, the chapter sheds light upon, how image data are treated within CNN and how the network learns useful representations of that data to finally arrive to a prediction.

3.1 Machine learning

Machine learning is a field within Artificial Intelligence, that is concerned with the construction of computer systems that automatically improve by learning from experience, that is, patterns and inferences made based on data. A more formal definition of machine learning was given by Mitchell (1997, p. 2): "*a computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T as measured by P, improves with experience E*".

Machine-learning systems are designed to look at many examples relevant to a task, to find structure in these examples that would allow the system to come up with a set of rules to automate said task.

In recent times, increase in the amount of data collected, and increase of computational capabilities led to the increase of utilization of machine learning in a variety of fields, such as healthcare, manufacturing, financial modeling, language processing, visual recognition, etc. Depending on the problem at hand, different machine learning methods can be employed.

According to Hastie (2013) a common taxonomy of learning algorithms can be made as follows:

- Unsupervised learning: to understand the relationship between variables or between observations within data

- Reinforcement learning: to find an optimal solution given a set of sequential inputs

- Supervised learning: to learn inferences ingrained in the data, that can be used for prediction or estimation.

Unsupervised learning

One of the branches of machine learning is unsupervised learning, that comprises of problems where no supervision or aid is needed during the learning process. In other words, for every observation there is input data, however there is no associated response/labeled output, meaning that learning is done blindly in the form of inferences. (Hastie, 2013)

Unsupervised learning algorithms are often used to better understand the data at hand: by finding distinct classes of observations (cluster analysis) based on similarities and dissimilarities between observations; or by reducing the number of variables of the dataset to a smaller number of useful variables (principal component analysis) that can be visualized in a two or three dimensional space.

Supervised learning

Supervised learning, on the other hand, comprises problems of predicting or estimating an output based on one or more input variables. Supervised learning algorithms are used to learn the mapping function f that can translate input's measurements $x_i, i = 1, \dots, n$ into output or prediction y_i, that can be used later for prediction purposes.

The term "supervised" denotes that the process of learning is aided by previously known outputs; the learning algorithm makes predictions on the training observations that are continuously corrected until the algorithm reaches a desired level of performance. (Hastie, 2013) An example of a supervised learning problem would be predicting the price of an apartment (y) based on m² of an apartment x_1, number of bedrooms x_2, postal code x_3, year built x_4. The first step of the process in this example would be to collect training observations, which possess input values (x_1, x_2, x_3, x_4) for which a corresponding output (y) is known in advance. Furthermore, the

algorithm would be fed with these training observations to learn a mapping function f that best relates the four input variables to the output variable. Lastly, the learned mapping function could be used for prediction: to translate new input data to its corresponding output.

Generally supervised learning algorithms can be categorized by the type of output that is predicted: algorithms predicting qualitative output in the form of labels (such as a cat or a dog) are called classification algorithms, whereas algorithms predicting quantitative output (such as the apartment price example mentioned above) are called regression algorithms. The image detection problem that is the matter of this thesis would fall in the former category; where the solution would predict an output in the form of labels: whether a given facial image is real or fake.

3.2 Evolution of Neural Networks

Artificial neural network (NN) is a bio-inspired branch of supervised learning. While the name suggests that NN is a representative model of how human brain works, NN is merely inspired by elements of our current understanding of the human brain. (Chollet, 2018)

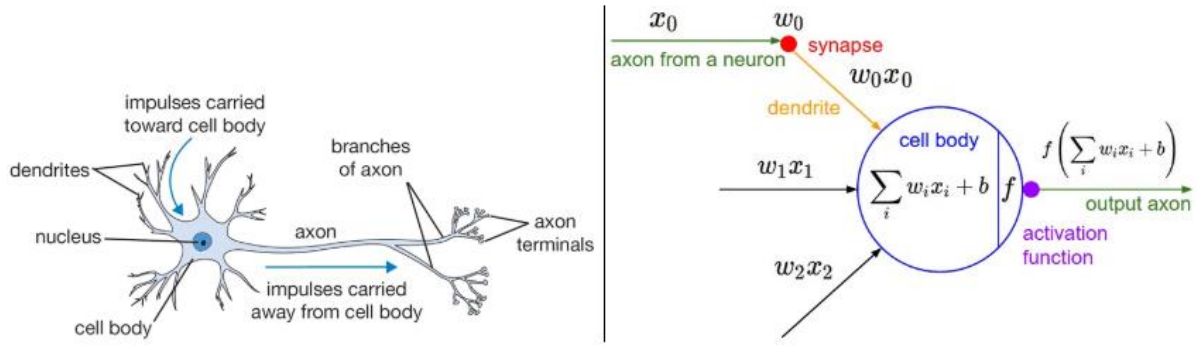


***Figure 3-1**: Comparison of biological neuron (left) to mathematical neuron (right) (source: Karpathy & Li, 2018)*

The first mentioning of NN is contributed to Warren McCulloch and Walter Pitts paper called "A Logical Calculus of the Ideas Immanent in Nervous Activity" in 1943. In their paper they proposed the first mathematical model of a neuron, that is still used in current NN models, although somewhat modified. Their neuron performed two actions, aggregated incoming inputs and made a decision on this aggregated information, thus becoming the output of the neuron. (McCulloch

& Pitts, 1943) While as the first implementation of an artificial neuron it was revolutionary, it also had its drawbacks. The input could only be a value of either 0 or 1, and the neuron´s decisions were only of two states "all-or-none" that were determined by a threshold value (e.g. given a threshold of 0.5, any value above the threshold would return 1, while below the threshold would return 0). Furthermore, all inputs were treated with the same importance, not allowing to differentiate between the usefulness of certain inputs (Pattanayak, 2017), which meant that inputs did not carry weights or biases.

The latter problem was solved by the introduction of the Perceptron in 1958 by Frank Rosenblatt. The project started with the idea of creating a human brain analog which could be used for analysis. It led to the creation of the Perceptron model for binary classification problems. In this approach compared to McCulloch and Pitts´ neuron, the inputs received by the neuron were not restricted to being binary but could take any positive or negative value. (Rosenblatt,1958) Furthermore, Rosenblatt introduced a weight term that was used to differentiate between the importance of inputs, that could adjust sample by sample, thus giving the Perceptron the ability to "learn". However, significant limitations were identified to this approach, including, the fact that it could not solve problems where inputs cannot be linearly separable, as seen in the Figure below (Minsky & Papert, 1988). The lack of solution to the non-linearly separable problems halted the advent of neural networks until the introduction to multilayer perceptron.

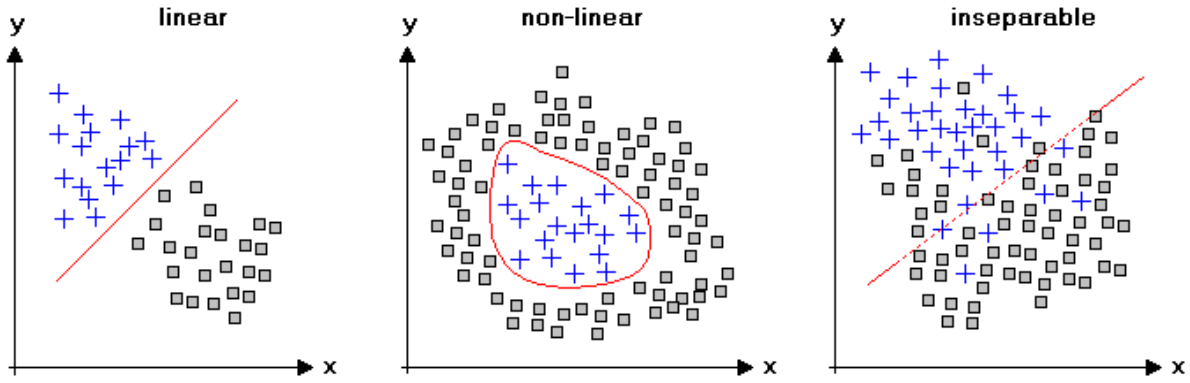


***Figure 3 -2**: Three categories of classification problems (source: Minsky & Papert, 1988)*

Finally, NN was revived by Geoffrey Hinton and his team in 1985 as they came up with the backpropagation method to learn multi-layered problems. It enabled to solve non-linear classification problems (Rummelhart, Hinton & Williams, 1986). NN reached is final form as we know it today by the name of deep learning in 2006. (Hinton, Osindero & Teh, 2006) The main

limitations of this technique were its tendency to "overfit" training dataset which could lead to a weak performance in predicting more distinct test data. It was solved soon after by introducing random dropout technique for train data (Hinton et al., 2012).

3.3 Explanation of Neural Networks

Figure 3.3 shows the various elements that NNs are made up of, which together aid during the training process to learn how to transform input data into predictions or class scores. The network is made of a series of layers that are stacked in sequential order and map the input data, using weights, into predictions. The loss function based on the predictions and the true targets (often referred to as ground truth) computes a loss score that evaluates how well the NN performs in comparison to the expectation. Lastly, the loss score is employed by the optimizer to know the extent to which the weights of the NN need to be adjusted. The following paragraphs will cover these elements in detail to provide an understanding of the inner workings of NN.

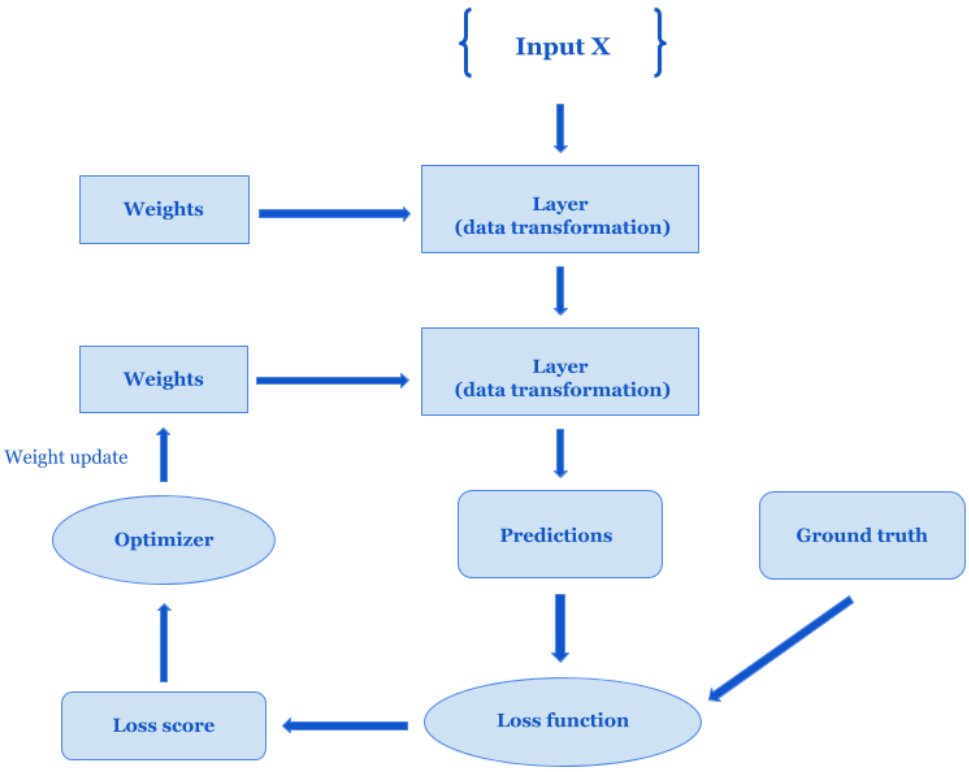


***Figure 3 -3**: NN elements and their dependencies (adapted from: Chollet, 2018)*

3.3.1 Layers

NN is comprised of layers that are the basic data structure in NNs (Chollet, 2018). Layers can be categorized into three groups according to their roles in the network:

- Input Layer that is merely passing data from the "outside world" to the Hidden Layer.
- Hidden Layer that takes input from the input layer or another hidden layer performs some calculations and passes the output to another hidden layer or the output layer.
- Output Layer that decodes the data from the hidden layer and transfers it to "the outside world"

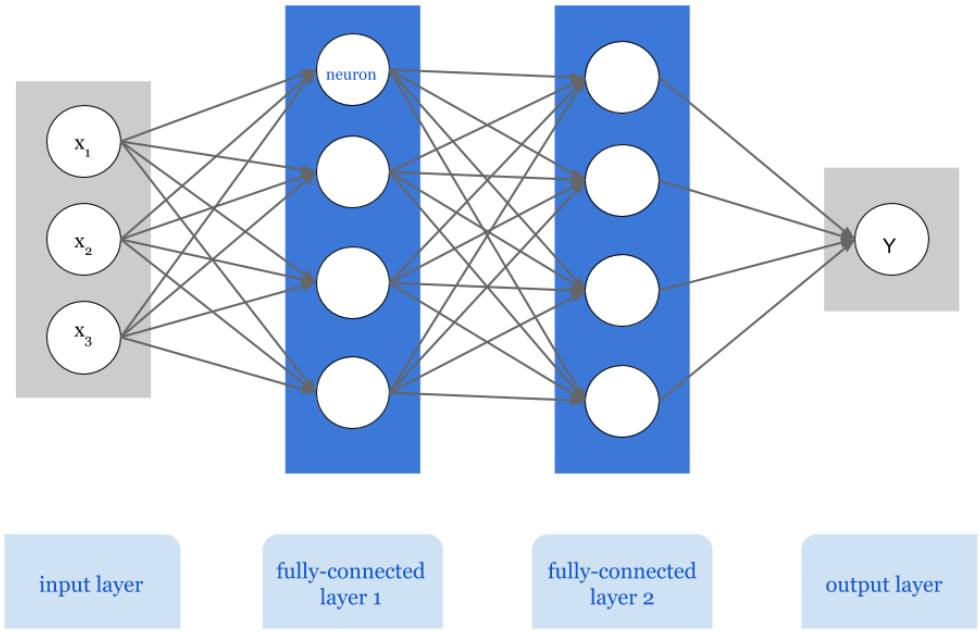


***Figure 3- 4**: Categorization of layers according to their roles in the network (adapted from: Karpathy & Li, 2018)*

An example of a simple NN with distinction of the layer types can be seen in Figure 3.4. The example shows a specific type of Hidden Layer, called Fully-connected Layer, which has all the neurons in the layer pairwise connected to the adjacent layers (but not to neurons in the same layer).

3.3.2 Neurons

The building blocks of NN are *neurons* (also referred to as nodes) that are organized into layers. As mentioned above, each neuron in one layer is connected to the neurons of adjacent layers, to transfer information within the network. In order to do so a neuron performs two actions: (a) aggregating incoming inputs and (b) applying the activation function. Looking at a single neuron inside a layer, the output z of the neuron is the result of weighted summation, between inputs $x_i, x = 1, \dots, n$ where $x \in \mathbb{R}^n$ and weights w where $w \in \mathbb{R}^n$ and the addition of a bias term b.

$$z = b + \sum_{i=1}^{n} x_i w_i \tag{3.1}$$

The second action performed by a neuron is to apply an activation function $f: \mathbb{R} \rightarrow \mathbb{R}$. That is a decision made by the neuron on the output (z) that produces the final output of the neuron y. Thus, the final output would take the form below (3.2).

$$y = f(b + \sum_{i=1}^{n} x_i w_i) \tag{3.2}$$

Activation functions are essential to NNs for two reasons, to speed up the training of NNs and to allow the network to capture non-linear interactions in the data (for example in cases, where a problem is non-linearly separable). Non-linear activation functions allow NNs to be networks, consisting of multiple layers. A NN consisting of layers of neurons with linear functions would be equivalent to a single hidden layer of neurons with linear functions, in terms of expansiveness. On the other hand, a combination of non-linear functions allows the network to be flexible and create a complex function that can deal with non-linearity in the data. (Ketkar, 2017)

Two non-linear activation functions have been commonly used in deep learning models, both of them being able to deal with non-linearity: the sigmoid function and the rectified linear unit (ReLU) function (Glorot et al., 2011). Sigmoid function has been present in machine learning in a variety of applications from logistic regression to basic NN implementations, due to its favorable curvature. Given an input of x , the output $f(x)$ can be written as

$$f(x) = \frac{1}{1 + e^{-x}} \tag{3.3}$$

Thus, the extension of the neuron from (3.2) using a sigmoid activation function will be

$$y = \frac{1}{1 + e^{-(b + \sum_{i=1}^{n} x_i w_i)}} \tag{3.4}$$

The sigmoid function has a smooth curve (Figure 3.5) that is also non-linear. In addition to that it transforms input data x_i to y where $y \in \mathbb{R}$ and $0 \leq y \leq 1$, which is ideal for probability estimation for example in the case of binary classification. Thus, it is still widely used as an activation function in the output layer of NNs, to estimate the probability of observations belonging to one class or another. However, the sigmoid function is not an ideal to be used in hidden layers, as the slope/gradient at either end of 0 and 1 are close to 0, meaning that even big changes in the input will result to almost no changes in the output of neurons applying sigmoid function, known as vanishing gradient problem (Karpathy & Li, 2018).

On the other hand, ReLU is simply a function that has a threshold at 0, meaning that the function is linear for values greater than zero, but also ads non-linearity as negative values always output zero

$$f(x) = \max(0, x) \tag{3.5}$$

Thus, the extension of the neuron from (3.2) using a ReLU activation function will be

$$f(x) = \max(0, b + \sum_{i=1}^{n} x_i w_i) \tag{3.6}$$

Compared to the sigmoid function this activation function (3.5) can output a true zero value,

$$f(x) = \begin{cases} 0 & if \ x < 0 \\ x & if \ x \geq 0 \end{cases} \tag{3.7}$$

meaning that sparse activations are possible (only activating neurons that have a non-zero output), which allows for faster training time of NNs. Furthermore in ReLU activation, the changes in the input value mean proportionate changes in the output value, consequently solving the vanishing gradient problem.

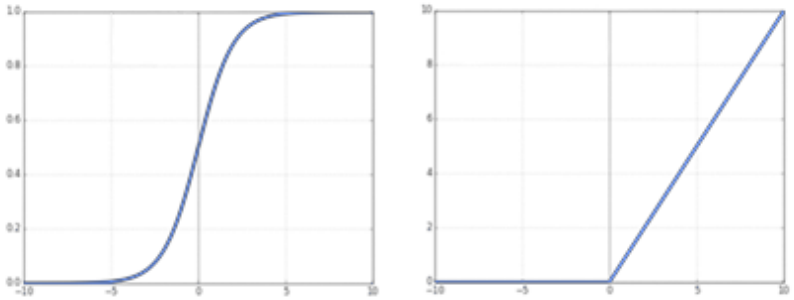


***Figure 3 -5**: Curve of sigmoid (left) and ReLU (right) function*

3.3.3 Weights and biases

Weights *w* and biases *b* are the trainable parameters of a NN. These parameters are continuously adjusted during the training process, to minimize the loss of the network. Each connection between two neurons of a NN has a weight. The purpose of weights is to indicate how strong is the connection between two neurons in the network. (Chollet, 2018) A strong connection (greater value of weight) amplifies the importance of the incoming information from a previous neuron, whereas a weak connection (smaller value of weight) diminishes the importance of a previous neuron´s information.

Furthermore, all neurons, except the input neurons, are connected to a bias neuron. The bias neuron is a special type of neuron that has a constant value of 1, and it has its weight that can be adjusted. Through weight adjustment, the bias neuron influences the activation function of a regular neuron, so that the regular neuron can output any value. Consequently, adding flexibility to the NN model.

3.3.4 Loss function

While in the previous sections, the actions taken by a single neuron were shown, there are multiple neurons in a NN. When the input data fed to the NN passes through the whole network of neurons (often referred to as forward propagation), the network outputs a predicted value e.g. a picture belonging to one class or another. The predicted value is compared to the “ground truth”, the

actual value of the input, to produce a loss value. The loss value is a simplification of the "good and bad aspects of a complex system" into a single number that can be used as comparison between candidate models (Reed & Marks, 1999). Ideally the closer the loss value is to zero, the better the model is at classifying samples in a dataset. The magnitude of the loss value directly impacts the adjustments made to weights, which will be covered in the following section.

Loss functions are employed to calculate said loss value. These loss functions can vary in how they penalize misclassification. Thus, the choice of loss function affects the weight adjustment and consequently the learning ability of a NN. For binary classification problems the most common loss function is binary cross-entropy (Ketkar, 2017). Supposing a dataset $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ where $x \in \mathbb{R}^n$ and $y \in \{0, 1\}$, and a generated model that predicts the probability of y given x. Furthermore given a denotion of this model: $f(x, \theta)$ where θ represents the parameters of the model, the idea of the binary cross-entropy function (3.8) is to find the right θ, that maximizes the $P(D|\theta)$.

$$-logP(D|\theta) = -\sum_{i=1}^{n} y_i * logf(x_i, \theta) + (1 - y_i) * log(1-f(x_i, \theta)) \tag{3.8}$$

3.3.5 Optimizers and Backpropagation

Learning in the case of a NN is handled by an optimizer. The optimizer is an optimization algorithm, that determines how the network should be adjusted based on the loss function. (Chollet, 2018) The goal would be to adjust the values of the weights up until they reach an optimum value that is a minimized loss of the network (the difference between the predicted values of the network and the "ground-truth"). Figure 3.6 shows a simple example, the loss in relation to the weight of a neuron, where the goal would be to find the weight of the neuron w that achieves the global minimum of the loss of the network $C(w)$.

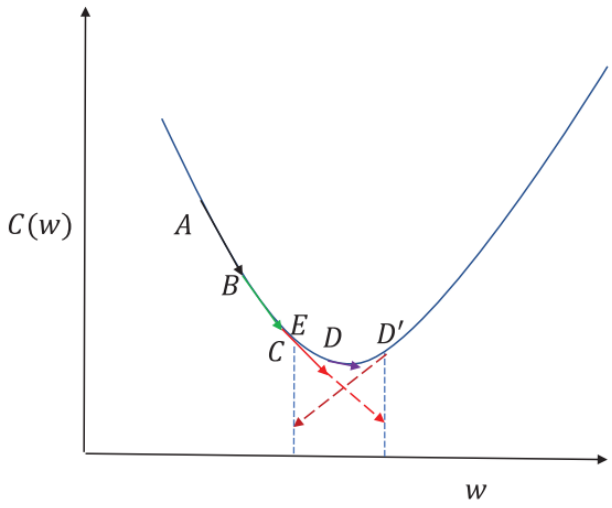


***Figure 3-6**: Gradient descent of a cost function with a single variable (source: Pattanayak, 2017)*

Said goal can be achieved through a technique called gradient descent, which is also used in various other machine learning algorithms other than NNs. The gradient descent algorithm iteratively: finds the direction in which the loss function has the steepest decrease and takes a "step" in that direction (by updating the weight), up until it reaches the minimum of the function. In order to find the direction, the gradient/slope of the function must be calculated using the derivative of the function with respect to the weight.

Given that a NN consists of a large number of neurons with their weights and biases, the loss of the network is an accumulation of the losses of all neurons' weights and biases. Hence to minimize the loss of the network, all weights and biases are updated based on their individual contribution to the total loss of the network. A backpropagation algorithm is used to find the contribution of weights and biases to the loss of the network. The goal of backpropagation is to compute the partial derivate ∂C/∂w and ∂C/∂b of the loss function C with respect to any weight *w* or bias *b* in the network.

3.4 Convolutional Neural Network

Similarly, to the NN covered in the previous sections, convolutional neural networks (CNN) are also made up of the very same elements: layers consisting of neurons that possess trainable

weights and biases. Neurons in convolutional networks still perform two operations: a multiplication followed by a summation function and a function that applies activation, often a non-linear function. Furthermore, convolutional networks also employ a loss function and optimization algorithm to aid the training of the network and to find the optimal parameters (weights and biases) that perform best for a given task.

3.4.1 CNN architecture

However, there are a few differences in comparison to the ordinary NN, namely that CNNs take input data in the form of images and the layer structure is different; the neurons in the hidden layers of the CNN are not fully connected to each other. In the following sections, only the unique elements of CNNs, that differentiate them from regular NNs, will be elaborated on to give a comprehensive understanding of how image is being transformed into prediction.

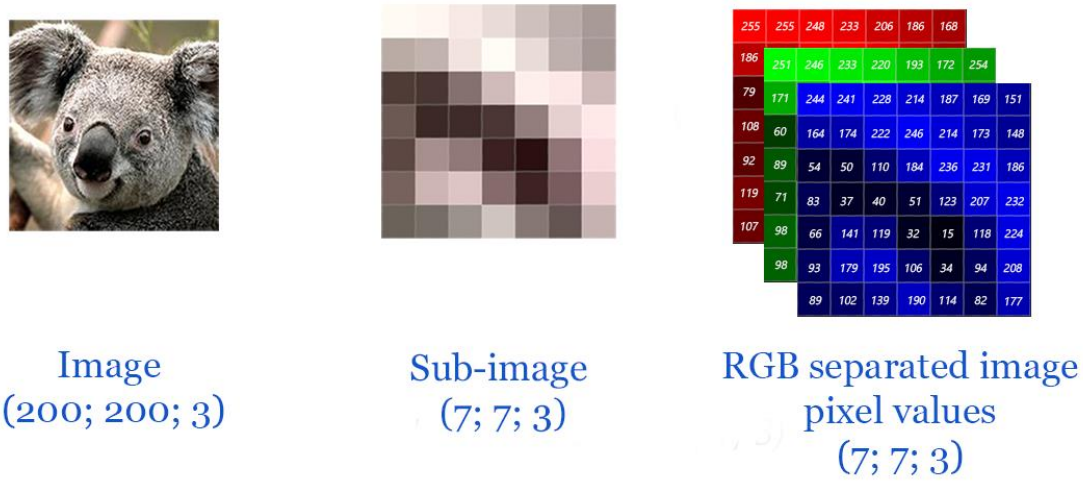


Figure 3-7: *Description of an RGB image with matrices of pixel values*

CNNs, as mentioned above, are most often applied to tasks e.g. image recognition, object detection etc. Thus, the input data is in the form of images that can be described as matrices of pixel values (between 0 and 255). Colored images are usually described using three color channels e.g. RGB, HSV etc. making the image 3-dimensional data. In Figure 3.7, the image is of height and width of 200; and since the image is RGB-based it has three channels (red, green and blue) that gives the image a third dimension of depth that is equal to 3.

In terms of pixel values, the sub-image (the koala´s left eye corner) can be described using three matrices of size 7 x 7 that correspond to each of the three color channels. By stacking the matrices, the result would be a three-dimensional structure known as a volume. Through a series of various layers, the convolutional network extracts the most useful features of the input volume, which can be used to classify the image into one of the predefined classes.

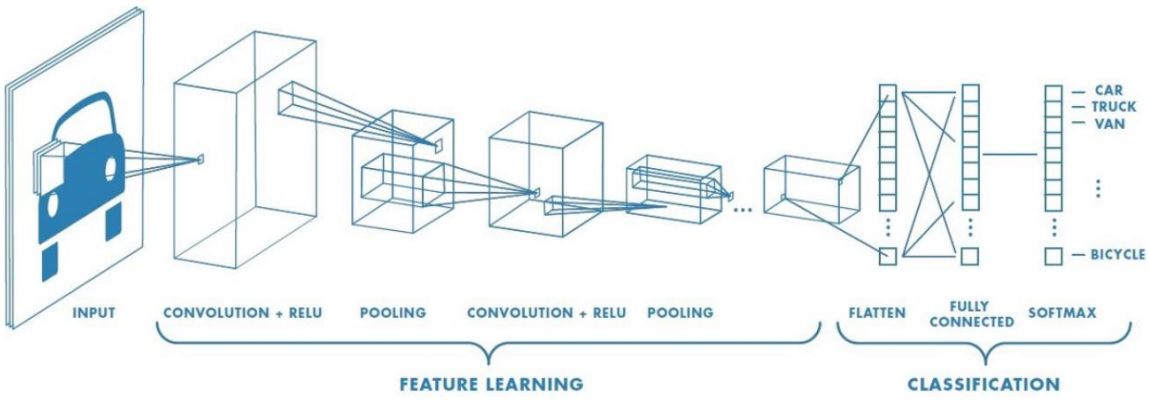


***Figure 3-8**: Architecture of a CNN model*

The architecture of a CNN (Figure 3.8) can be separated into two distinct parts, according to their function: a feature learning part and a classification part. In the feature learning segment of CNN, convolutional layers transform the pixel matrices (that is the input volume) using weight matrices (kernels) into feature maps, that separate the most useful features of the image. Rectified Linear Unit functions (ReLU) are applied to add non-linearity to data and to speed up the process of training by removing features of little interest to the CNN. To cope with the increased volume of data, the sizes of feature maps are reduced spatially using pooling layers, which help with both faster training of CNNs and with overfitting on the training data. The two layers: convolutional and pooling can be stacked upon each other any number of times, depending on the machine learning problem, size of images, size of the training dataset, etc., although some conventional wisdom and heuristics exist.

After extracting features using a series of convolutional and pooling layers, the final feature maps are flattened using a flattening layer, that transforms a volume into a simple vector. Thus, the

flattening layer connects the last pooling or convolutional layer to the fully connected layers that at the end compute class scores, i.e. whether an image is fake or real.

3.4.2 Convolutional layers

CNNs as the name suggests are in majority made up of layers that perform convolutions. A convolution is an element-wise multiplication of an input volume (a 3-dimensional data structure) with another volume called kernel, resulting in a matrix (a 2-dimensional data structure).

Neuron connectivity

Neurons in convolutional layers are organized alongside three dimensions: width w, height h, and depth d. The number of neurons n that can fit in a convolutional layer depends on the input volume:

$$Input\ volume\colon\ w_1 \times h_1 \times d_1 \tag{3.9}$$

and the hyperparameters chosen: stride S, no. of kernels K, kernel width/height F and padding P, before training the network. Given these parameters the output volume,which also describes the number of neurons n in a convolutional layer, would look the following:

$$w_2 = \ h_2 = \frac{w_1 - F + 2P}{S} + 1 \tag{3.9}$$

$$d_2 = K$$

$$Output\ volume\colon w_2 \times h_2 \times d_2$$

These neurons rather than being fully connected, as in regular NNs, are locally connected to each other. Each neuron in a convolutional layer connects to a local region of the input volume. The local region is a matrix called the receptive field of the neuron (R) and is equal to the kernel size of a layer (R = F x F). This connection is local alongside the width and the height dimension of the input volume, however it extends to the entire depth of the input volume as shown in Figure 3.9.

Figure 3.9 shows the connection between an input volume and a convolutional layer. The two blue-colored neurons highlighted are connected to only a small region (R = 3 X 3) of the input volume alongside the width and the height axis, however to the entire depth (d_1 = 3) of the volume. As mentioned above the kernel size and equivalently the receptive field is a hyperparameter chosen before training.

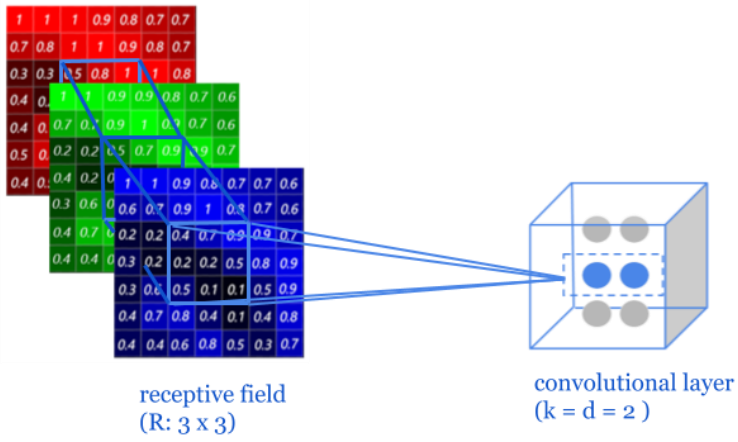


***Figure 3-9**: Receptive field and connectivity of a neuron*

The two blue-colored neurons highlighted are located in the same position along the width and the height dimension, but different positions in the depth dimension (different depth slices) of the convolutional layer. Neurons with the same spatial position but different depth position are tasked to transform the same region of the input volume. Given that each depth slice has its own kernel, the number of kernels determines the depth of the layer (d_2 = K). Since the two blue-colored neurons are in different depth slices and employ different kernels, they activate differently based on the kernel used. As an example, in the Figure 3.9, one of the blue neurons would perform a convolution to determine if the region contains a vertical edge, while the other neuron using a different kernel determines if there is a diagonal edge in its receptive field. Lastly neurons that are in the same depth slice, i.e a blue and a grey neuron, make use of the same kernel, however for a different region of the image.

Kernels

Convolutional networks also employ trainable weights that transform input volume from one layer to another. In the case of CNNs weights are organized into weight matrices that are called kernels. Kernels are constantly updated during the training of CNNs based on the loss of the network.

Figure 3.10 demonstrates a diagonal edge detection kernels that a CNN could reach during the training of the network. These kernels are small matrices, in the example in

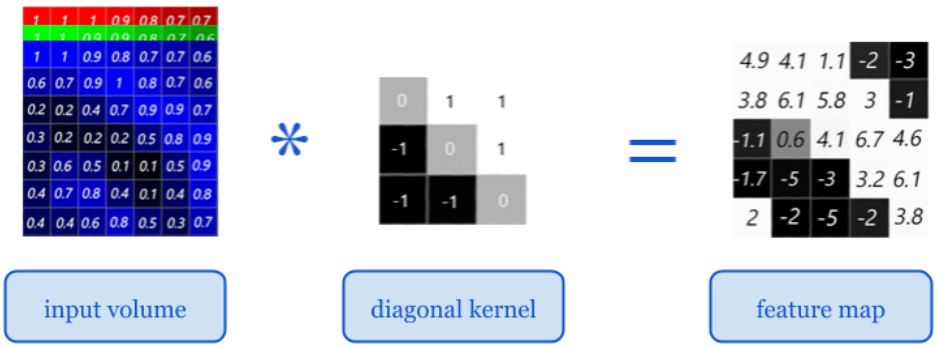


***Figure 3-10**: Feature detection using a diagonal edge detector kernel (note: for visualization of 3 dimensional volumes, depth slices are stacked)*

Figure 3.10 of size 3x3x3, indicating a kernel width and height equal to the receptive field of a neuron and a kernel depth equal to the depth of the input volume. Kernels are used to transform the input volume by detecting features that are specific to the kernel e.g. edges, color blobs. The resulting matrices are two dimensional and are called feature maps. After stacking together the resulting feature maps, yet another 3 dimensional volume is created. Said output volume, can be used as input volume to other convolutional layers.

Convolutional operation

A convolution is an element-wise multiplication between an input volume and a kernel, resulting in a feature map that is part of the output volume. The resulting feature map gives the activations of the kernel at each region of the image. In other words, the feature map indicates in which areas of the input volume there are patterns "similar" to the kernel. The patterns become more abstract with every consecutive convolutional layer, as Karpathy wrote: "the network will learn filters [kernels] that activate when they see some type of visual feature such as an edge of some orientation or a blotch of some color on the first layer, or eventually entire honeycomb or wheel-like patterns on higher layers of the network" (Karpathy & Li, 2018).

The figure below demonstrates how the first unit of the feature map is calculated. In the example the three-dimensional volumes are separated by depth slices for easier demonstration. The convolution between the input volume and the kernel can be summed up in the following steps:

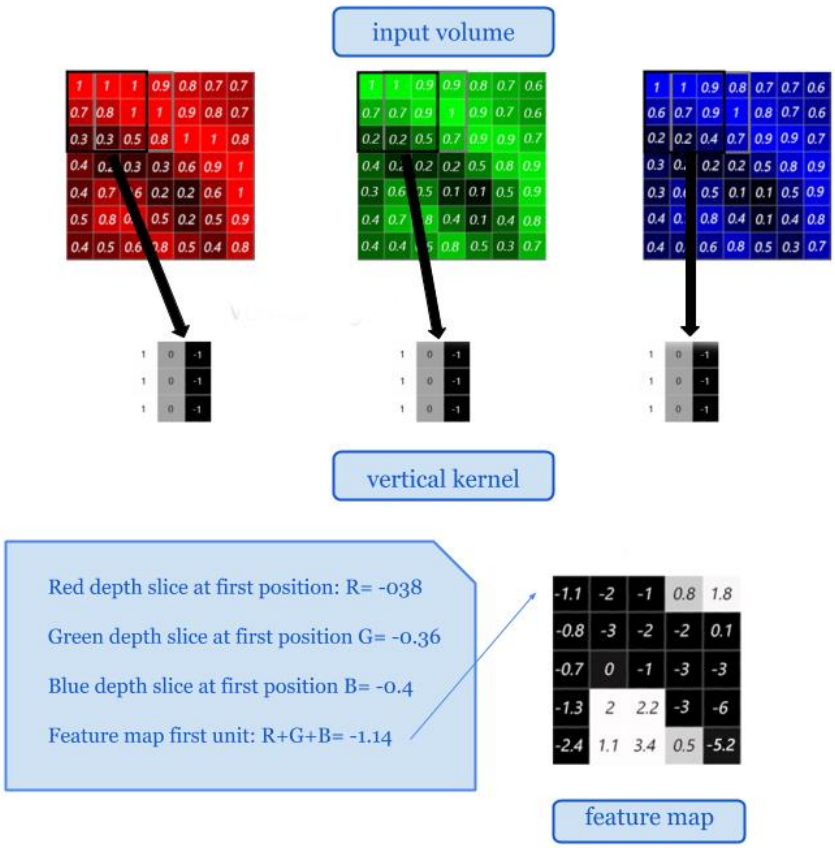


Figure 3-11: Kernel operation/ element-wise multiplication. (note: for visualization of 3 dimensional volumes depth slices are shown in a row)

1. Place the kernel on the top-left corner of the input volume
2. Element-wise multiply the values in that area of the input volume with their corresponding kernel values
3. Sum the results of multiplication (and add the bias term) to obtain the top-left value of the feature map
4. Slide the kernel with the value of stride S to the right (upon reaching the right edge, move to the next row on the left)
5. Repeat steps 2 and 3 until all areas of the input volume are convolved.

3.4.3 Pooling Layers

In between convolutional layers, it is common to insert pooling layers that reduce the spatial size of the feature maps in the network (Karpathy & Li, 2018)By reducing the spatial size the CNN

needs fewer calculations to converge, consequently it becomes faster to train. Additionally, the reduction in the number of parameters in the network helps to retrieve the most dominant features, thereby controls the overfitting in the network. The trained network thus becomes more generalizable to new images. The pooling operation is similar to a convolutional operation in that input data are transformed using a kernel that is "sliding" through each depth slice of the input volume. However, pooling layers do not contain any parameters, thus element-wise multiplication is not performed in these layers.

Figure 12. demonstrates how a max-pooling kernel with width and height equal to 2 and a stride of 2 reduces the spatial size of a 4 x 4 x 2 output volume retrieved from a convolutional layer. The pooling operation as mentioned above only reduces the spatial size of the input volume, the depth remains constant. The pooling operation only takes the most dominant value in a 2 x 2 window that is covered by the pooling kernel.

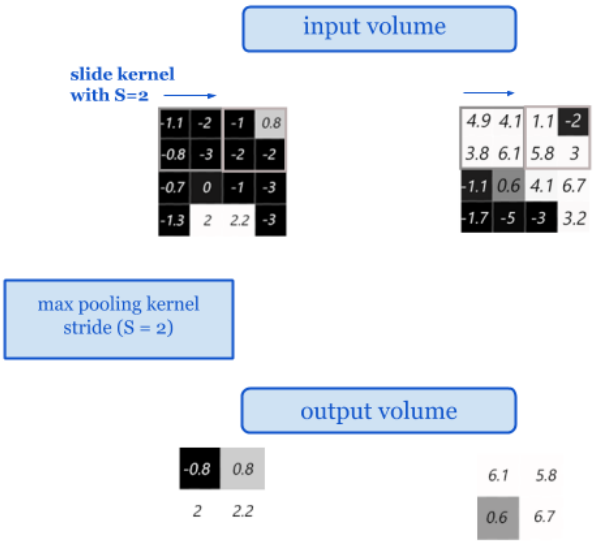


***Figure 3-12**: Example operation of a max-pooling kernel (note: for visualization of 3 dimensional volumes the depth slices are shown in a row)*

Pooling layers can be of two types: max and average pooling. While the former retains the maximum feature, value covered by the pooling kernel, the latter averages the feature values covered by the pooling kernel. However, both types result in feature maps that contain significantly less features. In the above example the output volume retrieved only 25% of the features that were inputted into the pooling layer, only keeping the maximum and most dominant features.

3.4.4 Fully connected layers

Fully connected layers in CNNs and regular NNs are alike. Neurons are fully connected between two layers, meaning that neurons from one layer are connected to all the neurons of the subsequent layer. However, in CNNs the use of fully connected layers is limited to the latter layers

of the network wherewith less features, fully connected layers offer a computationally cheap alternative to learning non-linear combinations among the features.

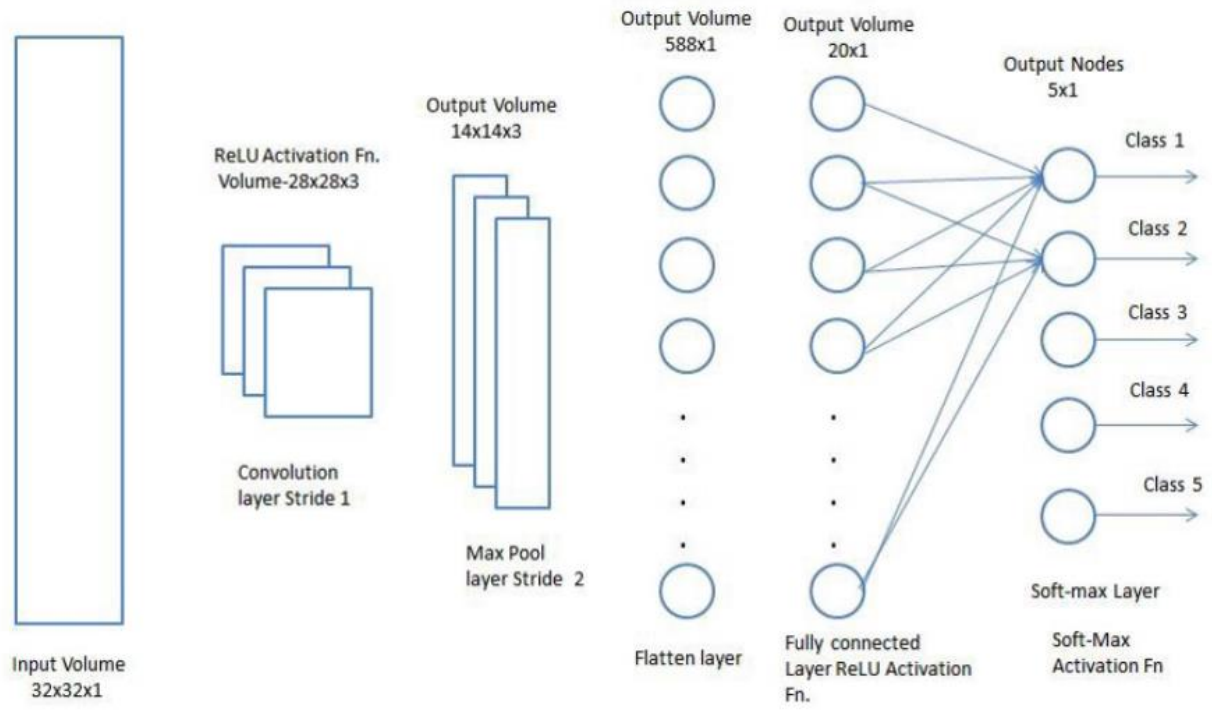


***Figure 3-13**: Macro view of CNN architecture including the flattening layer and Fully-connected Layer*

In order to connect the last convolutional layer with a fully connected layer all features of the convolutional layer that are three dimensional must be flattened to a single dimension (a vector) that can be used as input to a fully connected layer.

4.

Methodology

The following chapter serves as an introduction to the methods used for researching fake facial recognition. It starts by emphasizing the research approach taken throughout this project. Afterward, the data collection strategy is introduced, including selected datasets, their underlying characteristics, and what labeling was used for fake and real images. Privacy issues and datasets compliance with GDPR is discussed next. Afterward, an in-depth overview of all steps taken in the data processing pipeline of this project is presented, starting with the aggregation of raw input, e.g., face extraction, data augmentation, data normalization. We provide reasoning for chosen methods in the transfer learning part of the methodology and describe data transformation steps required by the adopted models. Next, we introduce the classifier training pipeline in which we examine how to implement the desired solution with different layers setup and how optimal hyperparameters were chosen. Then comes the part of how we evaluated the predicting power of the trained models. Lastly, we propose a comparison method to evaluate constructed artifact based on human-level performance.

4.1 Research approach

The scope of this research design is exploratory research within the Design Science Research paradigm. The purpose of exploratory research is to discover and describe an unexplained phenomenon, such as GAN and CNN application to fake news domain, and its correlation with elements in the given contexts. The phenomenon of Fake News identification through the analysis of visual content has very sparse research and high novelty factor, especially in combination with the usage of machine-learning, which is GAN and CNN application to this problem. This research intends to create an artifact capable of addressing this problem through the series of iterations on given datasets and artifact's validation against humans. (Briggs & Schwabe, 2011)

In alignment with Carlsson's (2005) paper on critical realism implementation in Information Systems (IS) within DSR, we have conducted this research from the critical realist standpoint. We believe that critical realism is the most appropriate philosophical approach towards the development and evaluation of IS artifact. As critical realists, we believe that reality is very complex; therefore, there are multiple of possible ways to develop IS artifact. Moreover, we believe that the solution to fake image recognition problem is solvable through actual practical and theoretical work and understanding the underlying methods used to design fake images in the first place. (Carlsson, 2005)

The theory development process of this project falls under the abductive approach. The primary reasoning for adopting this approach is the perception that there might be infinite number of viable means to fake images as well as to recognize those forgeries. Due to the complexity of the problem, we cannot afford to address every possible scenario. Therefore, we utilize the most reliable data available to us and observations based on the past attempts to deconstructed similar problems to develop the desired artifact. Abductive approach is excellent for developing and iterating plausible early-stage ideas, especially when resources within the domain is limited. As it is customary with abductive reasoning, we believe that if two the most frequent image forgery techniques can be successfully identified using the CNN model, then the problem at large is solvable with the same approach. (Saunders, 2016, p. 144)

Due to the immense practicality of the problem, we have chosen design science as our research strategy. The main benefit of the design-orientated approach is its applicability to practice and contribution towards solving a business problem (Saunders, 2016, p. 9). Design Science Research (DSR) is a generally accepted framework within the IS domain. While constructing a research strategy, we closely followed the framework proposed by Sharda (2010). Based on this framework, the DSR consist of iterative cycles. The general design cycle has five steps where each step yields either an outcome or reiterates back to previous steps. (1) The initial step is awareness of the problem, which outputs a preliminary proposal on how to approach the problem. (2) Theory building or suggestion step is for collecting knowledge on the topic, which is necessary to solve the problem and should output a tentative design of the solution. (3) Afterward, the actual development of the artifact takes place, which is the iterative process itself and commonly requires creativity and trial and error mindset in cases where the existing knowledge is insufficient. The output of this step is the best-performing artifact produced after many iterations. (4) Next step is the evaluation of the artifact based on empirical methods, including its comparison to existing artifacts. The result of this step is performance measures which can be any appropriate empirical evidence or logical proof. (5) Finally, the conclusion step, which is responsible for evaluating if

artifact solves the desired goal. If it succeeds, it outputs the final results. Otherwise, it reiterates back to previous steps for improvements of the artifact (Sharda, 2010, p. 27). Once all the steps in the general design cycle are finished, and feedback implemented, the actual process of the artifact development starts.

4.2 Data collection strategy

In order to create an artifact capable of detecting fake visual content, a vast amount of both real and fake images were needed that were altered using the two identified techniques: splicing and machine-generated. Until the time of identifying the research problem, there was no comprehensive dataset gathered of fake imagery in news articles in the magnitude needed to train a machine learning model on. Thus, alternate non-news related image datasets were found that were produced using the manipulation techniques mentioned above. These image datasets contained facial images, both legitimate and altered using splicing or generation. The combined amount of 22000 images were used from all datasets, with an equal number of images being either fake or real.

For this research to be completed a fraction of 3 publicly available datasets were used. The datasets in question were:

1. NVIDIA's StyleGAN dataset - Containing 100000 GAN generated fake facial images. (Karras et al., 2018)
2. Flickr-Faces-HQ Dataset (FFHQ) - Containing 70000 real facial images from Flickr. (Karras et al., 2018)
3. Yonsei University's Real and Fake Face Detection - Containing 2200 real and fake facial images. ("Real and Fake", 2019)

NVIDIA's StyleGAN dataset

Ian Goodfellow's creation of GAN method (Goodfellow, 2014) led to the creation of NVIDIA's implementation of unsupervised learning techniques for creating high-resolution (1024×1024) AI-generated human faces (Karras, 2019). Astonishing quality images were generated using GAN, which was originally trained on IMDB's celebA-HQ dataset in 2017 (Karras, 2018) and improved using Flickr-Faces dataset in 2018. From this dataset, we have taken 10000 computer-generated

fake images. According to NVIDIA to replicate their research and to train the entire GAN network on 100000 images, it would take 41 days using Tesla V100 GPU. Therefore, to make this research feasible on a virtual server with slower processing capability, it was required to reduce the number of images and their quality.

Flickr-Faces-HQ Dataset

A high number of real facial images were required to train CNN models for binary image classification. For the network to be able to differentiate between real and fake facial images, we also used a large Flickr-Faces dataset of high-resolution (1024x1024) images. People shared their images to the popular photo-sharing website Flickr, where they are made publicly available to everyone. Of the original dataset 10000, facial images of real people were taken to match the size of the StyleGAN dataset.

Yonsei University's Real and Fake Faces dataset

The Department of Computer Science at Yonsei University released a dataset on Kaggle inviting data scientists and enthusiasts to tackle the problem of fake images. These fakes images are different from GAN's in several aspects, namely technique and dataset size. First and most importantly, the images in the dataset were altered using the splicing technique with Adobe's Photoshop software. Splicing, by definition, means the cutting and merging of two or more different pictures into one. Secondly, unlike the StyleGAN dataset, this dataset is significantly smaller and comes with 2041 images of 600x600 resolution. All images in this dataset have certain parts of the image composited (spliced): left-eye, right-eye, nose, mouth. The dataset comes with three arbitrary difficulty levels of easy, medium, and hard, determined by human evaluators.

Data distribution

There is no official count regarding the distribution of facial images among these three datasets. However, based on our observations, we can conclude that those datasets include all age groups, races, both males and females. Images also contain people wearing glasses, hats, and different type of piercings. Hence, it should be represented across all groups, rather than biased towards a specific group of people.

Labeling data and final datasets

The final datasets to be used in our research were derived from the original datasets described above. A Photoshop dataset is created that is identical to the Real and Fake Face Detection dataset created by Yonsei University containing 2041 images, of which 960 were fake (using the splicing technique) and 1081 real images. Figure 4.1 visually demonstrates the composition of final datasets.

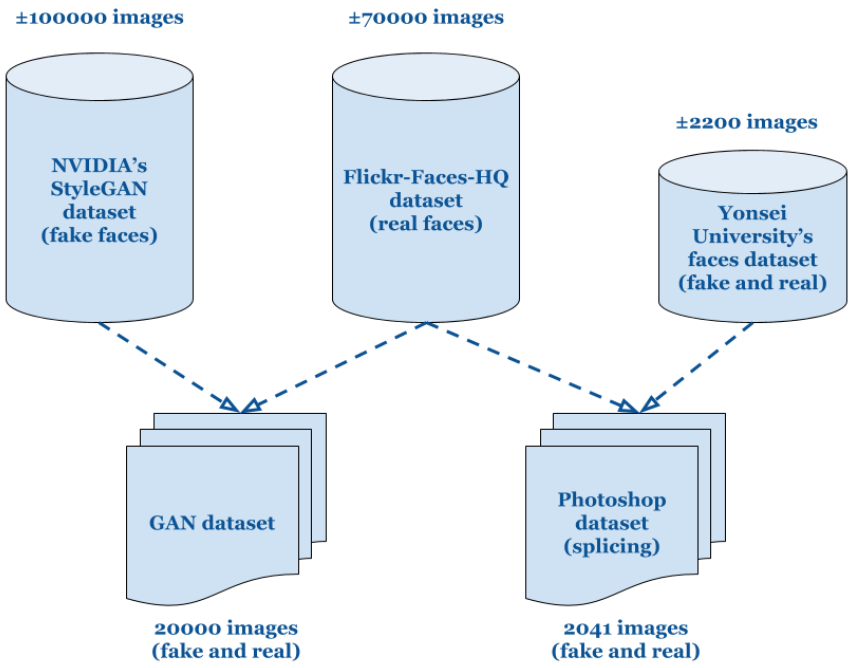


***Figure 4-1**: Origin of final datasets: GAN dataset and Photoshop dataset*

Figure 4.1 depicts the creation of our GAN dataset of 20000 images, where 10000 images were extracted from NVIDIA's StyleGAN dataset, that served as the class of fake images altered using a machine-generated technique (GAN). As for the remaining 10000 images that were of the class of real images, these were extracted from the Flickr-Faces-HQ Dataset (FFHQ). All extracted images were taken out randomly from the original datasets.

Privacy (GDPR compliance)

The data came from the openly accessible datasets which have been published either for educational purposes or as part of open source projects. In the case of GAN images, there were no privacy concerns given that they do not represent actual people. However, in the case of real images coming from Flickr dataset, all users have agreed with Flickr's Terms of Service and Flickr has announced that in compliance with GDPR all images can be removed on a request basis. Moreover, they are all publicly accessible and visible on Flickr's website. No personal data were assigned to the pictures, and no efforts had been made to identify individuals based on the pictures. Images of real human faces were not visualized in any part of the thesis, as they were only used during the training of CNN models.

4.3 Data processing pipeline

Our experiments were structured in a manner to create an iterative process where results of individual experiments can be used to improve the overall predicting model. In the research literature, such a process is commonly referred to as the "no free lunch" theorem, or in computer science, it is known as a "trial and error" strategy of problem-solving. Throughout the process, we kept coming back to the initial phase to repeat the entire process with changed parameters to reflect the problems we faced in the following steps. For instance, at the beginning of the project, we realized that our faked photoshopped images dataset might not be big enough. Therefore, we introduced the data augmentation part at the beginning of the process to increase the size of our dataset before preprocessing data. The visualization of the entire training protocol can be seen in Figure 4.4.

Data preprocessing

Given that the research is focusing on two distinct image alteration techniques, splicing and machine-generated, two datasets were created, each for separate models. Meaning that the Photoshop dataset was used to train CNN models to recognize fake images that were created using the splicing technique, whereas the GAN dataset was used to train other CNN models capable of distinguishing images that were altered using a machine-generated technique.

In order to assure that our algorithm does not become biased towards our training data (overfitting problem), both datasets were shuffled and divided into three parts. This step is essential in order to leave out a small sample of data for the final evaluation of the algorithm which has never been seen by the algorithm during the training step. Therefore, a division of 80/10/10 for GAN dataset and a 60/20/20 division for Photoshop dataset were used. 80/60% of images go to the training subset, 20/10% go to the validation subset, and final 20/10% go to the test subset respectively. This proportion for a division of data is commonly found in machine learning but might vary towards a higher percentage of training data in case of massive datasets. The choice of having only 10% of the GAN dataset withheld for testing purposes instead of 20%, was made since a test set size of 2000 images was deemed enough. Increased training set allowed to train more robust and consequently more generalizable CNN models on the GAN dataset. An essential part of this process was the randomization of the images into equal subsets (Figure 4.4, step 2). Furthermore, in the case of the Photoshop dataset to guarantee a proportional amount of easy, medium, and hard fake images in each subset for the evaluation part of the research.

Face extraction

In the next step after splitting the data but before feeding it to the CNN itself, the images were preprocessed. This includes several different forms of images transformation, including image resizing, flipping, applying color sharpening effects and cropping detected face areas out of the main image and vertically aligning cropped faces. For each original image, either faked or real, we applied the following chain of actions to preprocess it. We used C++ based dlib library tool for real-time face boundaries estimation based on 2016 research paper (Sharma et al., 2016). Face boundaries were used to extract only the part of images that contained faces (Figure 4.4, step 3). It was done to reduce unnecessary space of the image which did not include any distinctive face features, eventually making it easier for our algorithm to learn essential face features. Secondly, the library also offered tools for vertically aligning bounding box of the face area, making each image relatively comparable since mouth, eyes, and nose features were always vertically centered.

For instance, when cropping an image from GAN dataset (Figure 4.2) the vertically aligned bounding box of the face would be the final image which was inputted to CNN model.

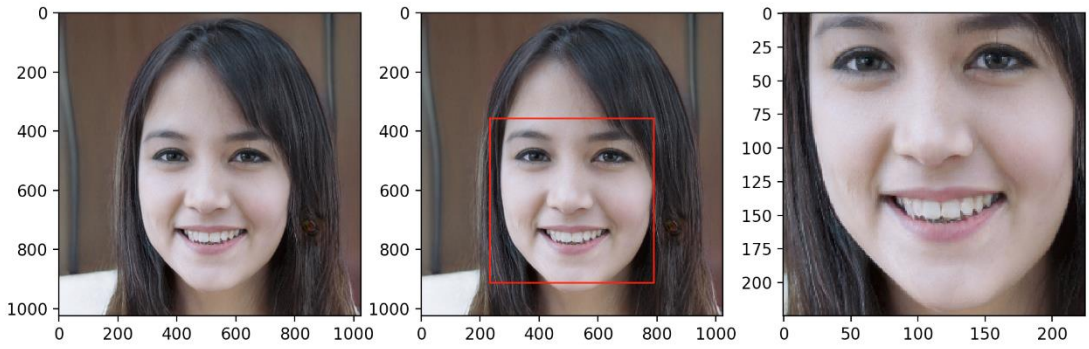


***Figure 4-2**: Face detection and cropping using facial landmarks*

Face cropping served another purpose which was discarding completely unfit images where the face could not be identified in the first place. Discarded images only accounted for less than 1% of the images since even side-view faces could be identified.

Following the face extraction and alignment procedures, images were resized so that each image would have an identical number of pixels, as the input layer of a CNN model has a fixed number of neurons corresponding to the number of pixels. Pictures in the Photoshop dataset were sized 600x600, whereas images in the GAN dataset were of 1024x1024 size. Since the face occupied area in images of the Photoshop dataset was estimated to be approximately 300x300, therefore extracted faces of both datasets had to be of uniform resolution, which meant that extracted face images were cropped down to be of size 300x300. Another reason for cropping down instead of scaling up was to avoid increasing the resolution of the worse quality images because lower resolution images would introduce dead pixels, over smoothing and artifacts when scaled above 100%.

For instance, the below 600x600 image from Photoshop dataset is already pixilated (Figure 4.3), which could potentially introduce noise to the CNN model, if after face extraction, the new image would be scaled up. Instead, after extracting the facial landmarks from the original image, a small part of the image was cropped to fit into a 300x300 size.

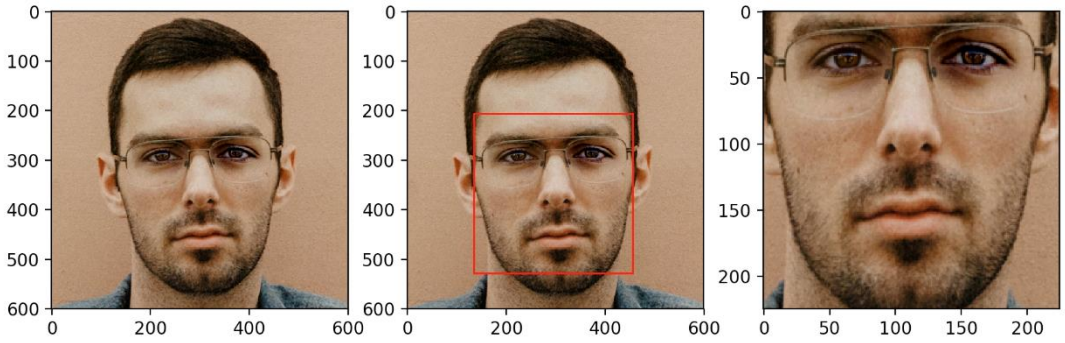


***Figure 4-3**: Example of a worse quality image and its resizing*

Data augmentation

Due to the shortage of images in the Photoshop dataset, we implemented several augmentation techniques to generate new images from the existing ones. At first, each fake image from the photoshop dataset was vertically flipped. By mirroring image, we had double this dataset and to avoid unequal distribution of classes we introduced additional real images from the FFHQ dataset. By extending our dataset, a significant improvement was observed in the predicting capabilities of our CNN models. We also tried other augmentation techniques e.g. adding additional noise to the images, random rotations, color blurs. However, they did not show any significant improvements in our experiments. Therefore, these augmentation techniques were not included in the final CNN setup.

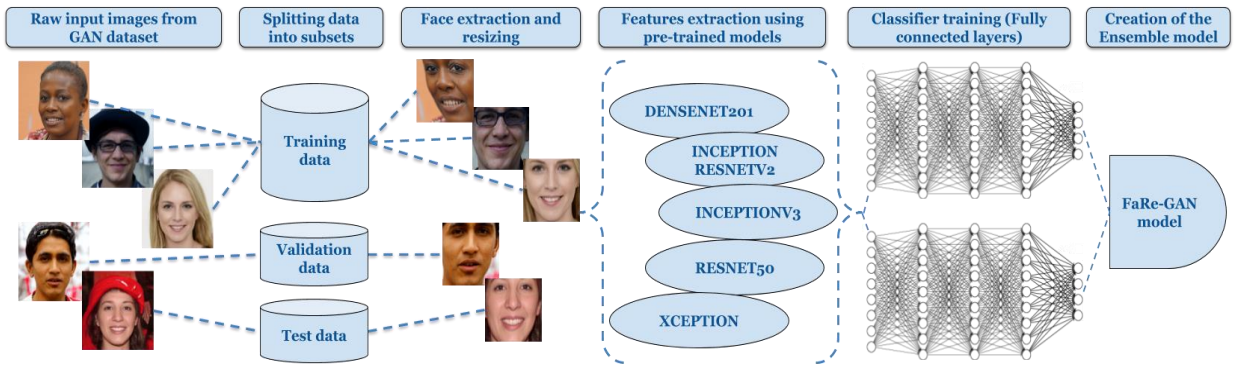


***Figure 4-4**: Visualization of the training protocol*

4.4 Transfer learning

While most ML solutions within visual recognition field can be developed from the ground up, it is often the case that previous solutions can be efficiently reused in development with a technique known as transfer learning. A useful characteristic of NN algorithms is their ability to transfer previously learned weights from even a distantly related problem to a new NN algorithm, to improve and accelerate the solving of the new problem.

The formal definition of transfer learning is the improvement of learning in a new task through the transfer of knowledge from a related task that has already been learned. Transfer learning on its own can be divided into an inductive transfer, Bayesian transfer and hierarchical transfer learning (Torrey, 2009) Our focus was on hierarchical transfer learning method where knowledge gained while finding a solution to a simple problem is passed over to solve a more complicated problem. For instance, annual ImageNet "Large Scale Visual Recognition Challenge" offers an opportunity for developers to create CNN algorithms capable of identifying 1000 classes of different things seen from images including animals, flowers, static objects, and even human faces. By using knowledge learned by these algorithms, we were already capable of identifying human or human's face. Therefore, we were using this knowledge to solve the more complicated problem of identifying whether a face within an image is fake or real.

4.4.1 Feature learning

In the feature learning segment of our CNN training (Figure 4.4, step 4), the network was supposed to find optimal weight matrices with which important features could be learned from the images in the two datasets. However, due to computational limitations and the size of the datasets (in case of the Photoshop dataset) we could not train NN models from scratch. It would have been extremely time consuming and very likely would yield unsatisfactory models in terms of performance. Thus, as mentioned above transfer learning approach was used, employing ImageNet competition-winning CNN architectures to extract useful features.

Feature Extraction

Feature extraction is an essential processing step in pattern recognition and machine learning tasks. The idea is to extract the key features of the object (for instance, from the image) and discard

less important ones. In our case, it was used to extract features from various ImageNet competition-winning NN algorithms, which have been trained on the entire ImageNet database. As a result, our model does not need to learn basic low-level features such as shapes, edges, or color blobs, since they were already learned by the lower layers of these winning NN algorithms. Moreover, feature extraction improves our own custom model's development time significantly since we only need to train our own model's top layers of NN while using the output of feature extractor as an input. In other words, it reduces the dimensionality of the initial set of raw data to more manageable groups for processing.

In practice, it is always a good idea to use feature extraction because depending on the size of the dataset and the similarity of the pre-trained model, the most appropriate way to reuse knowledge can be selected. Usually, in cases when the dataset is small, only the fully connected layers (which works as a final classifier of the problem) are removed, and the remaining lower and mid-level layers are used as-is. In our case, both datasets are relatively small, and images are partly different. Therefore, we have chosen to retrain only fully connected layers.

ImageNet pre-trained models

Since the beginning of ImageNet competition, people have been sharing their final classification algorithms. There is different motivation to reuse other models pre-trained weights. Firstly, it was noticed through the observation that the earlier features which are learned by the models tend to be more generic (e.g., edge detectors or color blob detectors). Therefore, by removing the top layers of the network (which learns specific objects classification), people can create well-performing models for custom tasks. Furthermore, pre-trained models can be fine-tuned to improve their performance. For our fake face recognition problem, we needed to remove the top layers of the pre-trained ImageNet networks and add fully connected layers on top of it for binary classification and re-train them. However, by reusing the more significant part of the network, it saved an enormous amount of computational time during the training phase.

Since the beginning of the competition in 2010 different winners appeared every year, bringing in new approaches and techniques into visual recognition field. Some model surpassed others and are still commonly used today. In this research, we used the following deep learning systems and models (hierarchical representations of layered features):

ResNet50 (He et al., 2015)

VGG19 (Simonyan & Zisserman, 2015)

VGG16 (Simonyan & Zisserman, 2015)

InceptionResNetV2 (Szegedy et al.,2016)

Xception (Chollet, 2016)

InceptionV3 (Xia et al., 2017)

MobileNetV2 (Sandler et al., 2018)

MobileNet (Howard et al., 2017)

DenseNet201 (Huang et al., 2018)

NASNetMobile (Zoph et al., 2018)

Data Normalization and Scaling

Since our data is used in combination with feature extraction from the best performing neural networks in ImageNet, we had to normalize and scale our data the same way as it was used while training those models. In general, features scaling processes are required by each image in the preprocessing part of machine learning due to two reasons: firstly, because raw data varies widely, and gradient descent-based algorithms converge way faster after scaling, which becomes crucial during training stage (Ioffe, 2015).

Different best performing models were trained using different machine learning backend frameworks, for instance, PyTorch, TensorFlow ("tf") or Caffe. Each of these frameworks are using slightly different scaling techniques. As a result "caffe" mode scaling generates values ranging from 0 to 255 or "tf" based scaling outputs zero centered values from -1 to 1 (also known as Z-score normalization) while "torch" mode outputs values in range from 0 to 1 (commonly referred to as Min-Max scaling when values are within a fixed range).5 For each ML application decision to normalize or standardize data should be decided individually. However, in our case, that part was defined since we were not training lower layers of the networks. Therefore, we had to use the same scaling values which were initially used to train those networks. Therefore, based on the requirements of the network, we selected scaling mode from "tf" "torch" or "caffe" during the data normalization part.

Second resizing

At this stage of the processing pipeline, we have extracted faces images of the size of 300x300 pixels. However, since we were reusing the weights of the lower layers of the ImageNet models, we had to use the identical input parameters, which in our case is image dimensions. Different

models were trained using different image sizes initially. Therefore, we needed to make a second resizing in order to match the format. In this step, models were divided into two parts: where older models like ResNet50, VGG16, and smaller architecture models like MobileNetV2 and NASNetMobile are using images resolution of 244x244 pixels. Therefore, we reduced our extracted faces to match the input size of those models. In cases of newer and bigger models like InceptionV3, InceptionResNetV2, Xception, we resized images down to 299x299 pixels to match the model's input requirements.

4.4.2 Classifier training

In the feature learning segment of our training, we have extracted the useful features within images using a transfer learning approach with the pre-trained models mentioned in section 4.4.1. The last convolutional layers of these pre-trained models were connected to fully-connected layers that were used for classifying our images into one of the two classes: real or fake. Thus, in the classifier training segment of our models, the extracted useful feature maps were used towards training a classifier that contained a number of fully-connected layers, where hyperparameters e.g. dropout and learning rate were determined using a grid search method.

Frozen layers

Low-mid layers, as explained above, tend to learn more generic features of an image. In the case of our NN models, these layers were frozen meaning that the weights of these layers did not change during backpropagation. If the weights could be changed, the risk of overfitting our network to the local scope of the problem would increase. Higher layers of the network are learning more specific features of the dataset; therefore, in case the datasets are very different, it needs to be either removed or unfrozen during the training stage. In our case both the low-mid and the higher layers remained frozen, given that there was a lack of computational power. Ideally, the higher layers could have been unfrozen to learn more specific features related to fake images, thus potentially improving the prediction capabilities of our models.

Fully-connected layers

All ImageNet models we used are using Sequential model architecture, meaning that all layers are set up one after another. To connect the convolutional layers of the ImageNet models to the fully

connected layers (Figure 4.4, step 5), the data needed to be flattened (transforms the format of the image from a 2d-array to a 1d-array) using a flattening layer; this layer has no parameters to learn. Fully-connected network structure usually consists of multiple dense layers, but the final output layer must have the same number of parameters (nodes) as the classification problem, which is being solved. Fully connected layers are at the top of an NN, serving as the final layers that at the end output a predicted value. If the problem is a binary classification problem, then the output of NN is usually 0 or 1 (as in our case). If it is a multi-class problem, then the output is the class number representing a specific object e.g. a cat, a dog or a car. As talked in the background, each NN layer needs an activation function. We used ReLU activation function in all dense layers of our fully connected network and Sigmoid function as the activation for our final (output) layer, which also acted as a classifier.

Dropouts

One of the most common problem people run into with relatively small datasets are overfitting or underfitting the model. Several regularization techniques can be applied to circumvent this issue. During our original iteration, we also ran into overfitting problems. Hence, we started using dropouts. A technique introduced in Geoffrey Hinton paper in 2014 which works by dropping a certain percentage of parameters randomly. During each iteration (epoch), a predefined number of neurons in the NN were set as inactive which nullified all outgoing connections of that neuron (Figure 4.5). It can be imagined as adding extra noise to the image. Researchers still debate how exactly dropouts improve the performance of the NN but are widely accepted as a standard technique to reduce overfitting and regularize data.

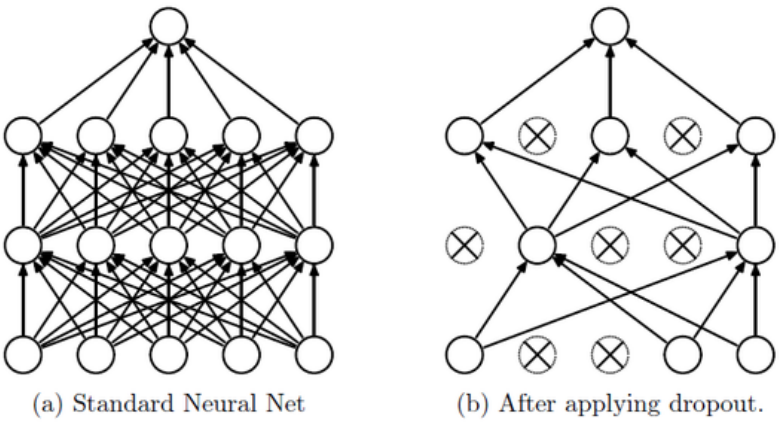


***Figure 4-5**: Effect of dropout on neuron activity (source: Karpathy & Li, 2018)*

Early stopping

Early stopping is a regularization technique meant for reducing overfitting (or more precisely stopping a training procedure before model overfits). It generally works one of the two ways: either by interrupting learning when learning loss stops decreasing for several epochs in a row or when validation accuracy does not improve for a certain number of epochs in a row. Due to the high number of models and different layers setup among those models in our project, early stopping allowed us to make significant improvements throughout the training process. On top of that, unfit models were abandoned very early in the process without finishing epochs during training.

Hyperparameters

The most important part of training NN is the network's weights adjustment. The maximum adjustment rate for weights during a single iteration is called the learning rate. As mentioned earlier, the stochastic gradient descent optimization process is used for updating the learning rate. The learning rate is a small positive hyperparameter which can be set between 0.0 and 1.0 (Goodfellow, 2016, p. 85). In return, it determines how quickly or slowly NN is learning. Meaning if the learning rate is too high, it will learn quickly but might end up with suboptimal weights and consequently, performance. In our experiments, different initial learning rate values between 0.0001 and 0.01 were tried. We achieved the optimal outcome/training time ratio with 0.001. However, that is only the initial learning rate, which changes dynamically based on the adaptive learning rate. Since the learning rate is hyperparameter, the optimal value needs to be identified using trial and error (Bengio, 2012). Therefore, we have chosen one of the most popular optimizers called Adaptive Moment Estimation (Adam), which includes stochastic gradient descent optimization process and an adaptive learning rate.

Loss function

The Loss function serves as a tool to determine how well a model is learning. As mentioned in the background part, high results deviation from the validation dataset's ground truth would lead to high loss value. Our project falls under the binary classification problems group. Hence, we have chosen to use binary cross-entropy as our loss function. Therefore, predictions which are confident and wrong are penalized the most by this algorithm.

4.4.3 Overview of training process

After hyperparameters are initialized training process using Keras framework can be started. Firstly, post-processed image data from GAN and Photoshop datasets were fed to a feature extractor. For each NN architecture from ImageNet competition winners, we extracted features, by running the post-processed data through all convolutional layers and extracting the modified data before fully-connected layers. We ended up with 20 models containing only the core features of these architectures. Training process separation into two parts, namely, features extraction and fully connected layers training, allowed us to reuse the same architectures for different fully-connected layer setups and hyperparameters. Finally, for each batch of extracted features, we trained our final models using a trial and error approach. We kept only one best performing model for each architecture based on validation accuracy. In cases where early stopping appeared before finishing training, we used model weights from the best performing epoch. We ended up having five models trained for classifying GAN images and four models capable of identifying fake and real Photoshop images.

4.5 Evaluating the models

To comply with the best practices of ML, we have never exposed our test datasets to our models during any iteration of the training process. After we were confident that we had well-trained models which were working well on validation dataset, we started evaluating our models using ensemble learning.

4.5.1 Ensemble learning

The reason for training many different models for the two datasets was to implement ensemble learning practice where different models' predictions (votes) would be combined in order to improve the final prediction (Kittler, 2000). The reasoning behind assembling the best performing models was that individually even the best performing models are making misclassifications. However, the models might have emphasized different features throughout the training process or have different weights. Thus, models might be making different errors. An

ensemble model however not only can improve overall accuracy but can also assure consistency for predictions on unseen data.

Conventional ensemble method techniques include voting, bagging, or boosting. Ensemble method is recommended to use for unstable learning algorithms where new data can easily change the output of the algorithm, or even the same training does not guarantee identical weights to be found (high variance is usually the case for small sample NN). All ensemble techniques rely on combining several base models and improving the prediction based on voting. The most straightforward approach is ordinary voting, where votes are weighted based on the accuracy of the model (Džeroski, 2009; Yaman, 2018). We used a voting technique in our research; hence, to make a final prediction on the test dataset, we combine all the votes from different best-performing models and use the outcome as the final prediction.

4.5.2 Evaluation measures

Both GAN and Photoshop test data were measured using an ensemble model to evaluate the overall accuracy of the model. However, to evaluate the results itself, we use several statistical models to describe our findings. Classification accuracies of our models were commonly referred to in order to compare how well given CNN models were performing. However, accuracies are only valuable if both classes are equally represented in the test dataset. Since we divided the data ourselves, we managed to get almost perfect 50/50 split of fake and real images in our test datasets. Furthermore, we used Precision, Recall, F-Score, Confusion matrix, and Area Under the Receiver Operating Characteristics (AUROC) score as metrics to calculate how well our two ensemble models are working.

To find out how well an algorithm is predicting between groups, one can make use of confusion matrices. In the case of binary classification problem, such as in the present research, it would be a table of 2x2, visualizing two dimensions of actual value and predicted value, resulting in 4 groups of True positives, True negatives, False positives, False negatives, that indicate mismatched predictions between the two dimensions. The importance of False positives and False negatives might differ based on the domain. Therefore, to find a balanced overview between the two F-Score can be calculated.

F-Score is calculated using the precision and recall values of an algorithm. Precision (4.1) is defined as the number of True Positives divided by the number of True Positives and the False positives. In our case, high precision would mean that the algorithm is performing well at identifying fake images (True positives). Precision is essential when False Positives need to be avoided. The highest theoretical value is one if all predictions of fake images were identified correctly, and no real images were identified as fake.

$$Precision = \frac{TP}{TP + FP} \tag{4.1}$$

Recall (4.2) is defined as the number of True Positives divided by the number of True positives and False Negatives. High recall value is prioritized in cases where False Negatives are highly relevant. In our case, the Recall theoretically could reach one if all predictions of fake images were identified correctly, and no fake images were identified as real.

$$Recall = \frac{TP}{TP + FN} \tag{4.2}$$

F-Score (4.3) calculates the Harmonic Mean between precision and recall. It is an excellent way to measure how precise and how robust an algorithm is; the higher the F-Score, the more balanced an algorithm is. F-Score can get values between 0 and 1.

$$F - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4.3}$$

AUROC score is the performance measurement to visually demonstrate how well the classification problem is distinguishing between binary classes. ROC curve is the probability of classes predictions and is based on the distribution of positive and negative predictions. AUROC score of 0.5 would mean that the algorithm is working as good as a random guess.

4.6 Hardware and software requirements

Due to the difference in operating systems and an abundant amount of dependencies required for this project, it is recommended to use identical setup to achieve comparable results. In order to replicate the study, the following hardware and software stacks were used.

External cloud computing server was needed to complete this project. It was provided by Copenhagen Business School and contained eight central processing units (CPUs) and 64GB of

random-access memory (RAM). On top of that, external hard drives were used to save all trained models, weights, and extracted features from the server to local storage, which takes 50GB of storage. Moreover, 40GB of additional storage is used by raw and processed images which were used to train the models.

The coding part was written using Python 3.7.4 libraries. External packages and dependencies are also needed. Among which the most important is Tensorflow 1.14.0 and Keras 2.2.4 frameworks. Moreover, the external C++ dlib library is needed to compile a tool needed for face boundaries identification. Github repository is used to store and share the code required by this project with the MIT license included. A copy of the datasets used throughout this research and the codebase can be found in Appendix C.

4.7 Benchmarking

Following the DSR paradigm as the last step in the design cycle, the artifact's performance needed to be evaluated; to determine whether there is a demand for such a tool. Given that there were no similar machine-learning-based models to benchmark on the task of false facial image detection, the judgement of the human perception seemed to be a reasonable threshold to evaluate the predicting performance of our two ensemble CNN models. To evaluate how the human eye would be able to cope with an identical task, we created a survey which contained 100 facial images (randomly selected from the same datasets which were used to test our models) with the following image distribution: 25 images extracted from the Photoshop dataset, 25 images extracted from the GAN dataset and 50 images from the FFHQ dataset.

To collect data from human evaluators, a snowball sampling approach was used on the Social Media platform LinkedIn. Snowball sampling in its essence is a sampling technique used to access hard-to-reach populations in an effective manner by identifying respondents who are willing to refer researchers on to other respondents (Miller and Brewer, 2003). While our respondents, namely news readers, were not necessarily a hard-to-reach population, the effectiveness of snowball sampling approach in terms of costs and collection timespan outweighed the innate bias that came with the approach. In a timespan of three weeks 38 people have fully evaluated the 100 images, while 5 people have evaluated a proportion of the images. LinkedIn was deemed to be an efficient platform for snowball sampling for two reasons: respondents could share the survey through their network thereby allowing for an effortless referral process and secondly LinkedIn

users are ranging from young to more senior adults (Smith and Anderson, 2018), which are the demographics that mostly rely on news website and Social Media to come across news (Shearer, 2018). However, in comparison to random sampling, this chain referral approach has a very important deficiency. Initial respondents and their refereed respondents have some sort of relationship thereby potentially violating the independency between respondents. (Miller and Brewer, 2003)

The survey was structured by first giving an in-depth definition of what fake and real images are in the context of the images within the survey. Further, all images were presented in randomized order, and interviewees were asked whether the image is fake or real. In order, to replicate the same conditions as for our CNN models the option of "I do not know" was not provided, since CNN models were also forced to always predict without meeting a minimum threshold of assurance. Definitions provided to the interviewees were as follows:

Real image:

(a) original image of a face taken with a photo camera;

(b) without any post-processing being applied in order to alter one or more parts of the face e.g., nose, eyebrows, mouth, or chin.

Fake image:

(a) a facial image of a non-existent person (created using a computer tool or drawn by a person)

(b) a facial image which has been edited with computer software, i.e. Photoshop in a way that changes one or more parts of the face i.e. by combining elements taken from several other images.

Lastly, to evaluate the respondent's ability to tell apart fake images, an aggregated confusion matrix and precision, recall and F-1 scores were calculated. Thus, the average survey respondent was compared to the combined Ensemble model. The outcome of the comparison is the accuracy rate for the CNN model and respondents given identical dataset with equal distribution of fake and real classes.

5.

Results

In order to develop the best performing fake image detection CNN models, the ten CNN architectures described in Section 4.4.1 were employed for feature learning, on top of which fully-connected layers were trained for classification. Throughout the development, the training protocol described in Section 4.4.1 and 4.4.2 were run appx. 100 times to develop candidate models, of which the best models were selected based on validation accuracy and loss. In this chapter the results of the best performing base models (Figure 5.1) for the detection of splicing and GAN alteration techniques will be presented, as well as the two ensemble models developed: FaRe-PS and FaRe-GAN. This chapter will present the result of the splicing and GAN detection models using both quantitative (accuracy, confusion matrix, F-measures) and qualitative means (grid visualization of misclassifications). Lastly to evaluate the performance of the artifact, the two ensemble models wil be evaluated in comparison to the human perception of detecting fake visual content, that was assessed through human evaluators.

DATASET	BASE ARCHITECTURE	FULLY CONNECTED LAYERS STRUCTURE
GAN	DENSENET201	419-D-1
GAN	INCEPTIONRESNETV2	628-544-D-21-16-4-1
GAN	INCEPTIONV3	68-1
GAN	RESNET50	63-1
GAN	XCEPTION	702-382-1
Photoshop	DENSENET201	76-1
Photoshop	MOBILENETV2	611-1
Photoshop	INCEPTIONV3	209-124-1
Photoshop	RESNET50	64-1

***Figure 5-1**: Fully connected layer structures of the best base models (note: D – dropout, numbers – no. of neurons in the layer)*

5.1 Detecting splicing technique

Several statistical analyses were conducted to evaluate the overall performance on the Photoshop dataset. Firstly, to have a generic evaluation of the performance of the CNN models, prediction accuracy was measured on the images within the test dataset. Afterward, to get a more in-depth insight into the individual groups of "Fake" and "Real" images, the precision, recall, and F1 score values were examined. Lastly, a confusion matrix was used to identify True Positive, False Positive and False Negative values required for the creation of the ROC curve to visually demonstrate the ratio between TP rate and FP rate in comparison to a random classifier.

After the training process, the top four best-performing architectures stood out based on their accuracy on the validation data. DenseNet201, MobileNetV2, InveptionV3, ResnNet50 outperformed other models during training and achieved 65%, 63%, 66%, 64% accuracies respectively on validation data. As a combined model, a custom Fake and Real model (FaRe-PS) was created using an ensembling technique. Figure 5.2 shows these models accuracies on the test data and the FaRe-PS predictions based on the votes from all four models. The accuracy of the FaRe-PS remained the same as the best performing base model - MobileNetV2. Having a voting system in place assured more consistent results, mainly because MobileNetV2 was the worst-performing model out of the four, based on validation data. The 65% accuracy of FaRe-PS, while still better than a random guess, is not enough for most use cases including, fake image detection. Further iterations and improvements are advised to cope with the identified shortcomings.

	ACCURACY	PRECISION	RECALL	F-SCORE
DENSENET201	64,54	0,65	0,51	0,57
MOBILENETV2	65,31	0,61	0,73	0,66
INCEPTIONV3	63,01	0,60	0,62	0,61
RESNET50	61,48	0,57	0,70	0,63
FARE-PS	**65,05**	**0,62**	**0,67**	**0,64**

***Figure 5-2**: Test dataset accuracies and F-measures of CNN models on the Photoshop dataset*

Confusion matrix was used to indicate the distribution of predictions among TP, FP, FN, and TN groups (Figure 5.3). The division was also essential for calculating Precision, Recall, F1-score, and AUC score for the later steps. The importance of TP and FP varies on a case by case basis. For instance, in the case of fake news identification, the aim is to maximize TP. Otherwise, identifying a fraudulent image as real would pose a risk of permitting potentially harmful content through without additional security checks.

		ACTUAL CLASS	
		Fake	Real
PREDICTED CLASS	Fake	122 (TP)	76 (FP)
	Real	61 (FN)	133 (TN)

***Figure 5-3**: Confusion matrix of FaRe-PS predictions on the Photoshop dataset*

Precision is a relevant metric designating the ratio between TP and FP values. More precisely, the proportion of fake images which were detected correctly. Arguably Precision is less prominent metric than Recall in fake news domain. FaRe-PS has higher Recall rate than Precision as seen in Figure 5.2. In other words, there is only a 33% chance of the fake image to be identified as real. F-Score combines the knowledge on both Precision and Recall into a single score. Extremely small F-Score would be a significant indicator that the algorithm is not operating as indented.

AUC-ROC Curve was used to demonstrate that the model can discriminate between binary groups of “fake” and “real” with better than random accuracy. Expected results of the random classifier marked with the red diagonal curve in Figure 5.4. The area under the curve (AUC) for the FaRe-PS achieved the score of 0,695 on the Photoshop dataset. The closer AUC score gets to 1 the more capable model is in distinguishing between groups.

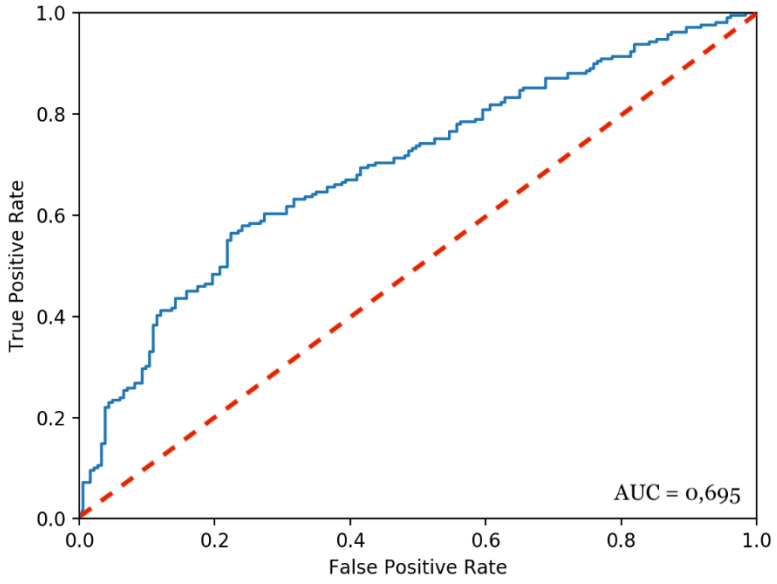


***Figure 5-4**: Receiver Operating Characteristic (ROC) of FaRe-PS on the Photoshop Dataset*

FaRe-PS predictions were visually demonstrated using a continuum representation of confidence in a 9x6 format (Figure 5.5) and 10x5 format (Appendix A). In Figure 5.5 each row in the continuum grid represents a 10th percentile of prediction's confidence. The top row contains only FP images with the confidence score of 0.9, or higher and the bottom row contains only TP images with the confidence of 0.1 or lower. Continuum grid serves as a tool to visually compare fake images in various confidence levels. For instance, the top row contains fake images where the FaRe-PS predicted that they are real. While the bottom row consists of only fake images identified as fake. The middle row illustrates visual features about which model is the most indecisive and predicts one of the classes with low confidence.

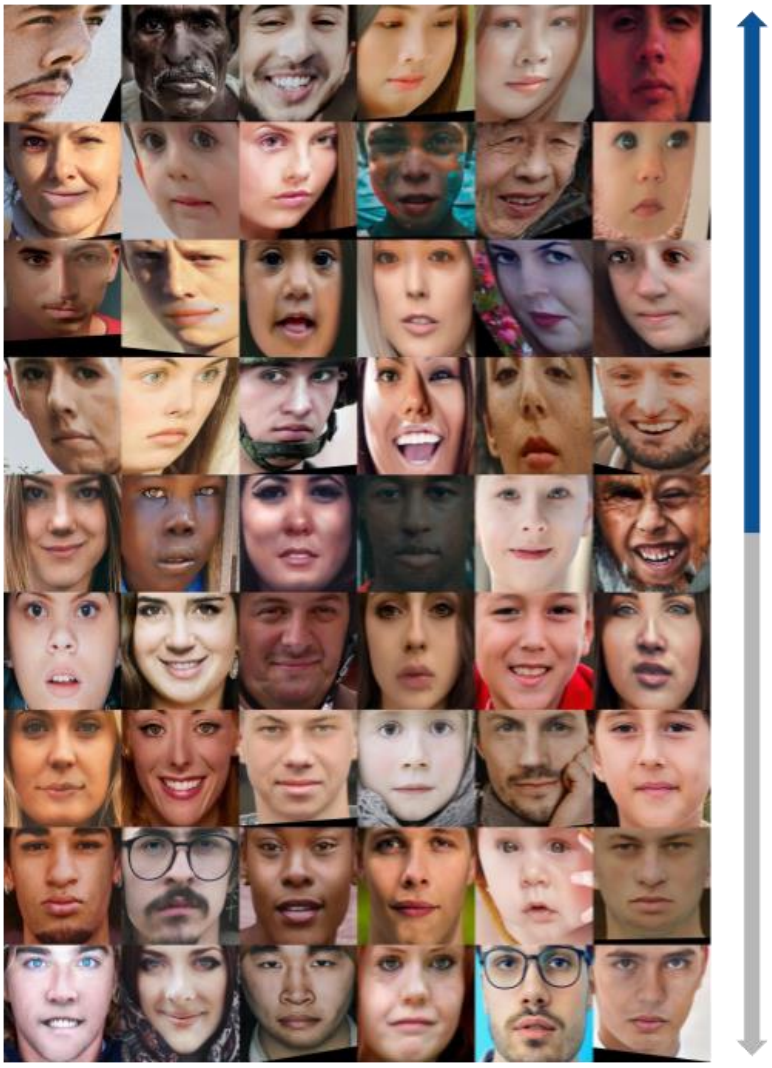


***Figure 5-5**: Continuum representation of confidence on fake images of the Photoshop Dataset. Each row represents a 10th percentile of confidence. Top row: False Positive with very high confidence (0.9 – 1); Bottom row: True positive with very high confidence (0 – 0.1)*

Several primary shortcomings of the FaRe-PS model can be recognized based on the assessment of Figure 5.5. The continuum grid shows that the model is biased towards side view faces. Half of the images from the top two rows are side-view faces, while none of TP on the bottom rows appear to be side-view. Moreover, a large portion of misclassified images have only moderately open eyes. Finally, in a significant number of misclassifications with high confidence alterations made are more apparent for the naked eye than in the majority of correctly classified with high confidence. Photoshop dataset consisted of fake images with one or more alteration made in eyes, nose, or mouth area. Continuum indicates that a vast majority of features learned by the FaRe-PS model do not correspond with the "faked" areas of the image. The most straightforward solution for improving the existing model in the future would be the expansion of the dataset. Alternatively, image division by areas of importance, e.g. eyes, mouth, nose, should be adequate to address most

issues seen from the continuum grid and is an improvement left for further iterations of this research.

5.2 Detecting GAN technique

The evaluation of the models' performance on detecting the GAN technique comprises of almost identical steps to the ones stated for detecting the splicing technique. The principal distinction prevails in trained models itself and which of them are used to establish the FaRe-GAN model. Additionally, a significantly larger GAN test dataset (2000 images in comparison to 400 images in the Photoshop dataset) permitted the production of more extensive continuum grid to visualize learning outcomes.

	ACCURACY	PRECISION	RECALL	F-SCORE
DENSENET201	92,92	0,95	0,93	0,94
INCEPTIONRESNETV2	92,55	0,93	0,93	0,93
INCEPTIONV3	91,44	0,94	0,91	0,92
RESNET50	94,36	0,95	0,95	0,95
XCEPTION	91,21	0.95	0.89	0.92
FARE-GAN	**96,81**	**0.98**	**0.96**	**0.97**

***Figure 5-6**:Test dataset accuracies and F-measure of CNN models on GAN Dataset*

Another remarkable distinction was that the best-performing architectures were partly different for GAN dataset in comparison to Photoshop dataset. The most plausible justification of such behavior is the nature of the images itself and the underlying heterogeneity in methods which were used to produce fake images. The five best performing models distinguished themselves from the remaining CNN architectures during training on the GAN dataset. The models in question were DenseNet201, InceptionResNetV2, InceptionV3, Resnet50 and Xception with 92%, 94%, 91%, 94% and 92% accuracies respectively on validation data. Figure 5.6 displays how each model and the FaRe-GAN model performed on test data. Due to the symmetrical distribution of

classes in test data, accuracy as a measure can be used as an objective measurement for estimating models' performance.

***Figure 5-7**: Prediction confidence distribution of ResNet50 model (left) and FaRe-GAN model (right). Each point denotes a test image.*

Contrary to the results on the Photoshop dataset, the ensemble model (FaRe-GAN) improved overall accuracy up to 96,81% on test images in comparison to the second-best base model, ResNet50 which achieved an accuracy of only 94,36% (Figure 5.6). Such effectiveness would be valuable for multiple use cases, including automated fake news tagging based on embedded images. However, the algorithm is only particularly capable of identifying whether a given face image is generated using the GAN technique or not. FaRe-GAN managed to improve Precision, Recall, and F-Score rates when compared to base models. The ensemble model successfully identified 98% of fake images as fake. Another striking trait of the FaRe-GAN model is that predictions shifted towards less confident in comparison to individual ResNet50 predictions (Figure 5.7). As seen from the image ResNet50 (on the left) often predicted either of the classes with 100% confidence, where FaRe-GAN's predictions (on the right) shifted towards the middle.

		ACTUAL CLASS	
		Fake	Real
PREDICTED CLASS	Fake	1143 (TP)	22 (FP)
	Real	47 (FN)	950 (TN)

***Figure 5-8**: Confusion matrix of FaRe-GAN model predictions on GAN Dataset*

The FaRe-GAN's excellent performance can be further seen when looking at its confusion matrix. Figure 5.8 shows that merely 47 real images were misclassified as fake. Low false-negative rate makes this CNN model fit for a task requiring a high Recall rate. Visually it was demonstrated using ROC curve (Figure 5.8). AUC achieved an astonishing score of 0.986. These results indicate that the CNN based model excels at detecting GAN generated fake facial images with a tremendous success rate.

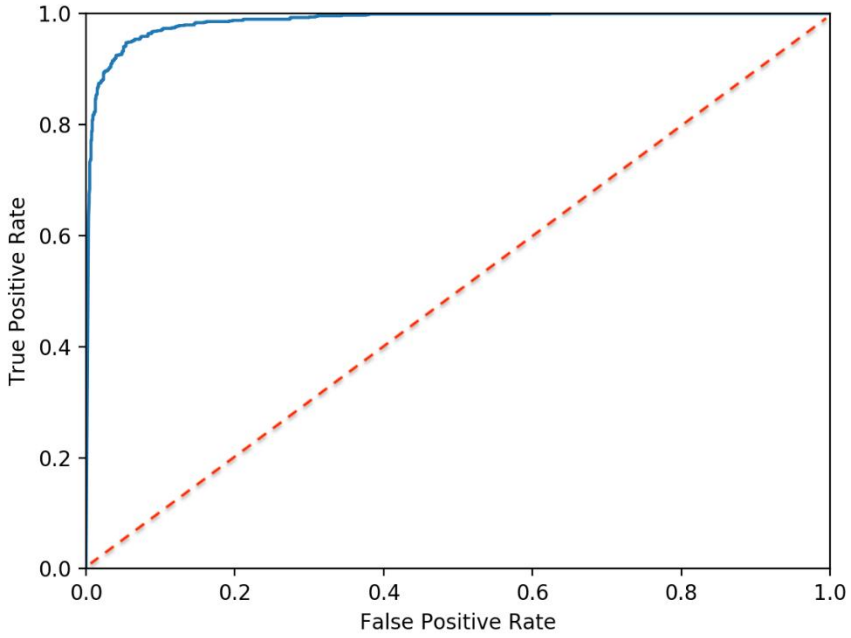


***Figure 5-9**: Receiver Operating Characteristic (ROC) of the FaRe-GAN on the GAN dataset*

Continuum analysis of photoshop dataset immediately revealed shortcomings of the trained FaRe-PS model. Applying the identical technique to GAN based 6x9 continuum grid (Figure 5.10) or even 50x5 continuum grid (Appendix B) did not reveal undeniable drawbacks of the FaRe-GAN model. However, it was also complicated to make claims regarding correct predictions. Most hypotheses are destined to remain disputable since it is troublesome to comprehend what learned knowledge lead to such high FaRe-GAN performance. Several statements appear convincing based on the continuum grid in Figure 5.10. The color scheme seems more homogeneous on the bottom rows, which means that the colors within images generated using the GAN technique are more likely to be monotonous. Manipulated images with GAN technique appearing in the

uppermost row holds numerous exceptions to this rule. Images containing red/colorful artifacts are more likely to be classified as real face. Finally, altered images using the GAN technique that were predicted as fake with high confidence (bottom row) seems considerably more complicated to detect for humans. On the other hand, the uppermost row included glaring irregularities and coloration blobs inside the images which would be manageable to distinguish for humans.

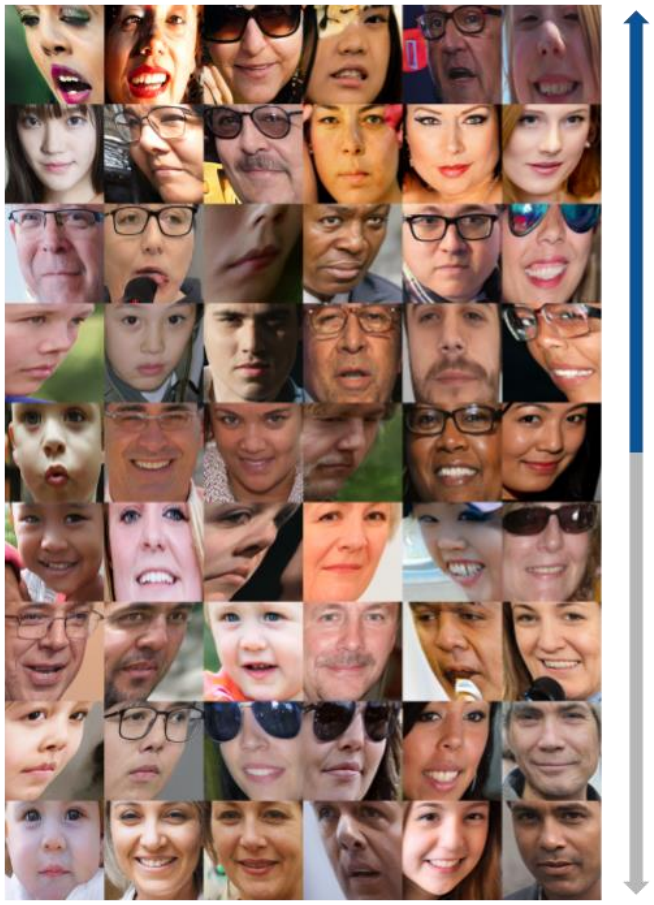


***Figure 5-10**: Continuum representation of confidence on fake images of the GAN Dataset. Each row representing a 10th percentile of confidence. Top row: False Positive with very high confidence (0.9 – 1); Bottom row: True positive with very high confidence (0 – 0.1)*

5.3 Benchmarking

In order to evaluate the performance of FaRe-PS and FaRe-GAN in comparison to the human perception of detecting fake visual content, 43 people have been tasked with evaluating 100 random images. The survey contained 50 altered images from Photoshop and GAN test datasets in equal proportions. Additional 50 unmodified images came from Flickr face dataset. The premise of the investigation is to verify to what extent individuals are competent in identifying

fraudulent visual content. Figure 5.11 shows the results of the study summarised in a confusion matrix where the predictions of all evaluators are featured in an aggregated form.

		ACTUAL CLASS	
		Fake	Real
PREDICTED CLASS	Fake	31 (TP)	9 (FP)
	Real	19 (FN)	41 (TN)

***Figure 5-11**: Confusion matrix of an average survey respondent on the 100 images*

Aggregated results showed that humans are relatively good at identifying real images. Evaluators achieved an 81% accuracy on real faces dataset. The slight drop in performance occurred when evaluating spliced images. Humans could identify only 65% of images from the photoshop dataset as fake. On identical images, the FaRe-PS model achieved a comparable accuracy of 68. Results imply that the detection of spliced images was a moderately complicated task for humans and computers alike. The evaluators experienced an even sharper drop in accuracy rate when evaluating GAN altered images where they identified only 57% of the images correctly. It came as a big contrast to the FaRe-GAN model which achieved an accuracy of 96% when voting on fake images altered using GAN technique. Such a decisive difference in performance is a relevant indicator that a CNN algorithm can be superior to humans in identifying GAN generated fake images.

	ACCURACY	**PRECISION**	**RECALL**	**F-SCORE**
HUMANS	71%	0,76	0,61	0,68
FARE-PS + FARE-GAN	**79%**	**0.77**	**0.82**	**0.80**

***Figure 5-12**: Accuracy and F-measures of an average survey respondent in comparison to the FaRe models*

Figure 5.12 supports the claims made above. Both humans and FaRe models had almost identical Precision rate, but the ensembled models had significantly better Recall rate. Overall F-Score also demonstrated a convincing gap between Humans and the AI in favor of computers. The survey serves as an indication that FaRe models can already outperform humans in fake visual content

identification. Therefore, they have a high potential and usability for tasks which require quick and precise image classification, such as false news identification. It might become even more critical in a case where GAN generated images become more widespread among fake news producers.

6.

Discussion

Evaluation of the machine learning model's performance is a daunting task. The first complication is asserting that test datasets expose all the variables of the desired task's complicated surroundings. Another concern is designing a testbed which captures the essence of the task that can, later on, be compared to the level of human perception. In the scope of this research, the detection of alteration techniques yielded inconsistent results. FaRe-GAN model performed exceptionally well on a test dataset to the extent of being an integral part of a real-time system. Even at its current state FaRe-GAN model would be able to classify nearly 97% of all faked facial images that had been altered with the GAN technique. On top of that, GAN based facial images yielded the worst results when evaluated by respondents. Humans were only slightly better than a random chance at identifying GAN based fraudulent faces. Such a distinct disparity in performance on the same dataset signifies the need for analogous CNN models in assisting humans when it comes to fake facial image detection.

On the other hand, the FaRe-PS model does not function as successfully as its GAN counterpart. While it still very closely resembles the performance of the evaluators, it falls short of achieving meaningful results at its current state. Interestingly, the FaRe-PS model was more reliable at classifying "hard" complexity manipulated images while it performed poorly on "easy" complexity images. On the contrary, humans identified "easy" images with higher accuracy than "hard" ones. The main takeaway from this iteration is that the FaRe-PS model and humans are highly plausible to misclassify photoshopped facial images. Therefore, an improved CNN model for such task remains a necessity. However, numerous fatal shortcomings of the current version of the FaRe-PS model were identified using the continuum grid of misclassified images. Hence, the insertion of the supplementary data into the training process would certainly resolve some apparent concerns of the current model related to the orientation of the faces within images. In the current state the FaRe-PS model seems to make distinctions between unrelated features; whether the face in the image is side-view or frontal portrait. Additionally, proposed concepts of the image cropping based on the essential areas would be a logical offset position for succeeding iterations.

Another notable outcome of this research is the successful utilization of the ensembling method to develop combined models capable of solving the fraudulent face detection problem. The ensembling method provided additional benefits for the detection of both GAN and splicing manipulation problems. Regarding the Photoshop dataset, the FaRe-PS model increased consistency among results and reduced the number of marginal predictions, which were overly confident when base models were predicting certain classes. While this ensemble method failed to significantly improve accuracy on the images within the Photoshop dataset, it increased the accuracy by almost 2% on the test images in the GAN dataset, which is a significant gain considering the overall high accuracy of the base models. On top of that, it had also improved disparity of the predictions for GAN dataset compared to the best performing singular ResNet50 model. Overall, the introduction of the ensembling technique allowed developing a more robust model capable of classifying fake images with reduced variance.

The GAN dataset comprises images of extraordinary quality which, as shown in the benchmarking part is capable of deceiving people 43% of the time. The success of it, as described in the original research on GANs (Goodfellow, 2014), lies in the framework of "discriminator" and "generator" models. The discriminator is a NN trained to differentiate between binary classes, while the generator is responsible for creating new images until discriminator misclassifies the image. Thus, it might seem counterintuitive that CNN ensemble model would be able to classify images with such accuracies since those images were already “purified” to the state where the discriminator failed to distinguish it from the original class. Nonetheless, as demonstrated in the continuum grid of GAN dataset predictions, GAN manipulated images might contain specific patterns which make them vulnerable to detection. Though most of the patterns remain debatable, it is hard to argue regarding lack of color diversity in GAN dataset, which might be enough information at the pixels’ level for the FaRe-GAN model to grasp on. However, GAN's ability to improve means that whenever a CNN model capable of detecting GANs appears, a more magnificent quality image can be produced by the generator, making GANs detection a never-ending mission.

Lack of reliable results of FaRe-PS model can be associated with a high diversity of fake images. Though, technically all images in photoshop dataset adopted a splicing technique, the application of such method differs based on individuals who apply it. In other words, manual intervention into an image requires varying levels of expertise and is profoundly plausible to generate a significantly unique product. Heterogeneous data in combination with a relatively small sample of data, makes the Photoshop dataset a challenging task for the FaRe-PS to comprehend.

6.1 Possible justifications for misclassifications

The first step towards improving classification models is to understand why misclassifications occur. NN based models are notoriously challenging to evaluate and debug. Hence, researchers have implemented all kinds of procedures seeking to shatter the contents of the black box of NNs, including the invention of sophisticated external software specializing in evaluating NN models (Odena & Goodfellow, 2018). While such software might be indispensable for massive-scale problems, it is possible to take a more qualitative approach towards debugging of NNs. Visualizing FP and TP images as an arbitrary sized image grid based on the confidence of predictions creates a visual tool of evaluating why errors arise at each confidence level.

Based on the continuum representation of the misclassifications made on the Photoshop dataset, several fundamental concerns require addressing in future iterations. According to the continuum grid, the FaRe-PS model has learned that images containing side-view faces are more likely to be classified as real images. That is partly caused by the fact that it is more prevalent to manipulate frontal portrait faces than side-view faces, which is also apparent in the Photoshop dataset. Expansion of the training dataset is a primary instrument for improving the ensemble model tasked to detect alterations with splicing technique. Additionally, another likely cause of misclassification is a binary division of images to fake or real. By perceiving an entire fake image as fake introduces additional noise into the training process. For instance, in cases where the splicing happened to noise area only, the mouth area and eyes area should be treated as unaffected. Consequently, it requires separating each image in the training dataset into tinier subareas, effectively conceiving a model fitted for ascertaining which section of the face was altered using computer software.

Contrary to images in Photoshop dataset, images in GAN dataset do not contain any areas with "real" information. This feature of GAN manipulated images partly explains the tremendous accuracy of the FaRe-GAN model. After an initial analysis of GAN classifications based on continuum representation, it is hard to describe patterns between FP and TP predictions. However, even for the unattended eye, the visual difference between the top row and the bottom row is distinct. The reason for such an effect is the smooth colors transitions in the TP (bottom row) and higher variance of colors in the FP (top row). FaRe-GAN has learned that images with distinct facial colors (face cosmetics), irregularly dyed hair, obstacles in front of the face (microphone) will belong to real images. These observations are a matter of debate since the total volume of FP images in the GAN dataset is barely a handful. The further extension of test dataset

would help to verify the above-mentioned observations. One approach to rectify these issues would be to exclude the background of the images. In the current state, the background of images altered by GAN is smooth looking while real images are more likely to contain objects which provide hints for the FaRe-GAN model and distracts its “focus” from the face itself. Moreover, adding a training step on black and white versions of the images would reduce the model's reliance on the colors itself and instead emphasized the sharpness of the color. Prior mentioned techniques are sensible candidates for adjusting the FaRe-GAN model based on visual feedback from the continuum.

6.2 Suggestions for upcoming design cycles

On a broader scale, the automatization of fake news detections could benefit from implementing the artifact proposed in this research. FaRe models, which are the outcome of this research, could be considered as two parts of a working prototype for designing a full-fledged product as it is common within design science. There are several proposals which should be implemented in future iterations before final artifact capable of solving a real-life business problem.

In order to significantly alter the process of fake news detection, the final product should be capable of taking a proactive approach toward fake news identification, rather than being a mere response/feedback-based system. The system should preemptively warn users of questionable news articles before the target consumes the content of suspicious nature.

The FaRe-GAN model based on NVIDIA's dataset showcased consistently positive results in fake image detection, it is reasonable to believe that the model would benefit from the inclusion of deep-fakes dataset. Though deep-fakes are mostly the same as GAN based images or videos, however, they contain recognizable facial features of real people (Korshunov, 2018). Deep-fakes is the ideal tool for producing fake news or harassment videos and has already affected many high-profile celebrities. Therefore, transfer learning approach should help to extend the final artifact of this project towards the successful detection of deep-fakes based images or videos.

The proposed final product should also be capable of solving a multiclass classification problem. At the current state the FaRe models trained on each dataset make binary predictions whether an image is tampered with or not. However, with the inclusion of additional techniques which are lesser used to generate fake visual content and were not within the scope of this research, the final ensemble models should be competent of performing multiclass predictions based on various

forgery techniques. Besides covering a broader spectrum of techniques, the final version should not solely focus on facial data but rather on images in their entirety to be truly applicable to a real-life business problem.

Finally, to get a better comparison between the ensemble models and survey respondents, an improved benchmarking technique should be implemented. Users' feedback is an essential step to enhance the IS artifact in DSR domain. Therefore, the evaluation survey conducted within this research should be expanded to permit respondents to assess how confident they are with their prediction. Consequently, a direct comparison would be possible whether human-level perception on tampered images is similar to the confidence of CNN algorithm. Moreover, to resemble more realistic scenario images should be coupled with textual clues. Frequently, it is hard to make a definite distinction between fake visual and fake written content within the same source. Thus, both pieces of information should be presented together for evaluators.

6.3 Contribution to research

Generative Adversarial Networks and its application to computer vision has been in the spotlight for quite some time. Consequently, extensive research has been already done in this area ranging from GAN's application to unsupervised learning problems (Radford & Metz, 2016) to image-to-image translation and pattern recognition (Choi, 2017). New prominent techniques of GAN implementation and its improvements has managed to achieve tremendous results in the computer vision domain. However, virtually no research has been done regarding the detection of GAN altered images. If misused in fraudulent activities GANs have a devasting potential, especially in the advent of deep fakes and fake news in general. Therefore, this research is a novel attempt to create an affirmative action towards detection and preventions of spreading GAN based fake visual content.

The benchmarking part of this research supports the claim that people are highly inefficient of differencing GAN generated faces from real faces even at the current state of the technology. As pointed out by Gulrajani (Gulrajani, 2019), GAN is not a simplistic memorization or convergence technique of machine-learning due to its ability to generalize ideas. In other words, GANs do not only transform an image into another image from the training dataset but instead, they are capable of comprehending what set of traits the desired image requires. Therefore, Gulrajani's proposed GAN evaluation framework claims that GANs need to be evaluated based on the intended task. There are no similar machine-learning-based models to benchmark on the task of

fake facial image detection. Thus, human-level performance is the baseline for determining whether there is a demand for such a tool. The outcome of this research contributes towards establishing the baseline for fake content identification, through the FaRe-GAN model, which drastically exceeds human-level performance for the equivalent task.

Proposed FaRe models in this research devise a preliminary solution for detecting tampered images. Fake news domain is a relatively new subject in the context of fake visual content detection. While research on the topic of deployment of GAN manipulation in fake news and its detection is sparse, splicing method detection got a considerable amount of attention over the past years. While there are notorious attempts in detecting spliced images (Huh, 2018) and GAN generated images in general (Nataraj, 2019), there was little effort to composing a centralized model in the context of fake news. After the first iteration of this project, a combined ensemble model originated which intends in combating fake facial images from the two most common techniques. The first iteration of this project served as a proof of concept, that state of the art CNN models can be used to detect different image manipulation techniques. Further inclusion of different image tampering techniques used in the fake news domain should extend existing FaRe models in the future iterations.

6.4 Proposal for implementation

Combating fake news is essentially a quest for building potent news quality differentiation mechanisms that prevent the news landscape being flooded with misinformation, falsehoods, and fake news. Traditionally in offline media, editors and journalists were trusted with informing readers about events happening in conjunction with checking the veracity of information. However, in online news markets the dissemination of news has shifted towards Social Media platforms. These platforms, through employing news curation algorithms may feature less reliable news articles in a trade-off for more traffic and advertising revenue. Thus, in online media the functional role of fact-checking has to be transferred from journalists.

In online media, news quality differentiation mechanisms need to be put in action either (1) in the news dissemination process within Social Media platforms (2) or by levering the use of third-party fact-checking. Another potential option is to allow readers to determine the authenticity of news themselves, as people initially rely on their own judgment to separate facts from fiction within news stories (Tandoc, 2017b). However, readers have innate biases: they tend to place trust in articles that are within their own spectrum to confirm their beliefs (Flaxman, 2016); in the context

of Social Media they tend to trust both the content of a news article and the news organization more, if the article is shared by their trusted peers (Turcotte, 2015). Whereas they decreasingly trust Social Media platforms to help the readers distinguish misinformation. In such scenario the news quality differentiation mechanisms, such as a fake image detection algorithm, need to be placed either at the disseminators or in the hands of third-party organizations that have a more objective judgment of source and information within news articles.

Social Media organizations as disseminators of online news were in recent times reluctant to combat fake news, abstaining from limiting the freedom of speech. However, their role in fake news detection could potentially be enormous, given that during the fake news dissemination process these platforms are the sole means of carrying out preventive actions. Meaning an early detection of fake news prior to the spreading of such news and reaching their target audience. Preventive actions are especially important given that readers are more likely to consider news as valid when said news had already widely spread and readers' perception had become increasingly difficult to alter once they accepted the trustworthiness of fake news (Roets et al., 2017).

Early detection of fake news relies on a limited set of information that is either the source or the content of potentially fake news, as is in the case of a fake image recognition algorithm. To this end, a potential use case for a fake image detection algorithm would be its embedding within a larger automated detection system on Social Media platforms. Whereas the fake image detection algorithm given adequate training time is able to detect falsified visual content in the form of images, the falsification of images is still in its infancy, in the context of fake news. Up until recently, only a minor subset of fake news articles could be assumed to contain falsified visual content. Consequently, fake image detection algorithms would be inadequate to detect fake news articles as stand-alone solutions.

Ideally, the larger detection system on Social Media platforms would classify fake news based on a multitude of variables such as the source of fake news, psycholinguistic and syntactic cues within the text of the news content, as well as the authenticity of the visual content. For the detection system to be robust against the diversity of news content and to minimize misclassifications on both ends (fake news classified as legitimate news and vice versa), these variables have to be jointly assessed. The detected fake news articles would be presented to users in a flagged or a rating-based format, where readers are warned of posts containing fake news articles prior to engaging with the content of said posts. Thus, readers still would be the de facto decision-makers in the process, where they would have the choice to read or disregard said article. Furthermore,

from the perspective of Social Media platforms, a flagging or rating system would not limit their intentions of maintaining the freedom of speech on these platforms.

Alternatively, another potential use case for fake image detection algorithm would be as part of the third-party fact-checking process. The process itself consists of monitoring news sources through various communication channels, finding claims within the news content, determining the source of evidence to check the claims, and evaluating/rating the veracity of claims. Currently the majority of this process is a manual task at fact-checking organizations, however some parts of the process can be automated. Monitoring, for example, can be done for video sources using speech recognition or for online newspapers using web-scraping. However, checking the factuality of content within articles is one area, where manual fact-checking still needs to be employed (Babakar & Moy, 2016). To this end, a fake visual content detection algorithm can help reduce the amount of manual labor to be done. By automatically detecting alterations in embedded images, a subset of potentially fake news articles could be efficiently detected, leaving the efforts of manual fact-checkers to be expanded on fake news articles which contain claims that are hard to refute automatically.

7.

Limitations

The research presented above has faced several limitations at two stages of the process: at data collection and algorithm training. Although, one of the critical strengths of the paper lies in the performance of the FaRe models tasked to recognize machine-generated fake images, assessing patterns in misclassified images were close to impossible due to the small number of misclassifications. The 22 False Positives and 47 False Negatives out of 2000 test images were too little to conclude any significant patterns within misclassified images, that could be used for further improvements. Ideally to construct an even better performing algorithm the test dataset should have been extended to contain more test images.

Regarding the Photoshop dataset that contained images manipulated using the splicing technique, the trained FaRe-PS model did not reach a satisfactory performance due to the underlying limitations of the dataset. Given that machine learning solutions rely on considerable amount of training data to account for variance, the 2000 images collected by researchers at Yonsei University were too little and contained "noise" within the data. A large proportion of the fake images were frontal portrait faces, probably as these images are more comfortable to be manipulated, whereas a large proportion of un-modified images were side-view faces. As a consequence, the FaRe-PS model likely has learned to differentiate between frontal-portrait and side-view faces in the images. While arguably a better performance could have been reached given a larger training set size, the complexity of creating additional manipulated images in the magnitude of 1000s was neither feasible nor realistic within the scope of this research.

Furthermore, more precise machine learning models could arguably have been achieved by training the FaRe models from scratch rather than employing a transfer learning approach. Transfer learning approach was ideal for carrying out this research for two reasons: (1) the problem domain (fake facial images) was similar to what the original pre-trained machine learning models were trained to classify (faces), and (2) transfer learning is a computationally inexpensive approach fit for the limited resources provided to this research. However, given

adequate computational resources custom models could have been built from scratch, which are less prone to overfitting. The overfitting problem was especially apparent with the FaRe-PS model trained to classify alterations made with the splicing technique. The pre-trained machine learning models had more trainable parameters than training data points, making the FaRe-PS model prone to overfit on the training data.

Finally, there is a limitation to the data collection method chosen to assess the baseline performance of human perception on the task of telling apart altered images from legitimate images. Snowball sampling approach was ideal to find evaluators in a cost- and time-effective manner, but the use of chain referral on the Social Media platform, LinkedIn, potentially violated the independency between evaluators, which means that initial respondents and their referred respondents have had some sort of relationship, likely in terms of educational or professional background. As such the average evaluator was potentially more educated and more aware of image alteration techniques, in particular Photoshop modifications. However, given that the performance of the FaRe models surpassed the results of these "educated" evaluators, the average newsreader is likely to perform worse given the same evaluation setting. To definitively conclude on the superior performance of the ensemble models, a random sampling approach would provide a more generalizable result, however at the expense of increased surveying timespan and cost, both of which were outside the means of this research.

8.

Future research

Based on the characteristics of fake news identified and the results of this research, the following potential research paths could be undertaken to facilitate a deeper understanding of both fake news detection in general and in particular utilizing fake image detection:

1) While the thesis was limited to include only facial imagery for the detection of two common image falsification techniques, to advance fake news detection through the detection of fake imagery, future research should be extended to other image classes. Ideally, the detection of image alterations should be generalizable, so as an algorithmic solution would detect alteration techniques regardless of the subject within the image.

2) To this end, efforts should be expended to create a uniform fake news dataset for benchmarking purposes. To our knowledge, currently, there is no collection of fake news articles that is comprehensive enough to be able to accommodate the many ways fake news can be detected and to directly compare various detection methods e.g. detection methods relying on textual cues, visual elements, etc. The benefits of benchmark datasets can be highlighted in the current advancement in deep learning research. Research into deep learning algorithms heavily benefited from benchmark datasets such as the ImageNet classification problem. To the point where the state-of-the-art image classification algorithms were the result of the comprehensiveness of the benchmark datasets. Similar advancements in fake news detection research could be made given a uniform collection of fake news articles for which researchers could propose and compare their solutions.

3) Lastly, a potential research area within fake news detection research would be hybrid content-based approaches. Content-based detection approaches are superior to other approaches such as ones relying on user response, given that they allow for preventive

responses prior to the spread of fake news. Hybrid approaches that make use of both textual and visual data within articles could prove to be reliable solutions to the wide range of fake news articles that are currently spread online.

9.

Conclusion

In recent years the phenomena of fake news has attracted a significant amount of interest from media outlets, social media organizations, governing bodies and non-profit organizations. At the heart of the discussion the two objectives of interest are raising awareness among members of society and devising solution to the increasing problem that fake news is expected to become in the near future. Motivated by these discussions and the increasing phenomenon that fake news is prognosed to become, the purpose of this thesis was to design a preliminary artifact that can detect fake news articles through the detection of embedded fake imagery using machine learning. While existing work has typically addressed fake news detection through means of either text content, user response to articles or the source of articles, we have investigated detection by focusing on images embedded within articles.

To this end, an artifact was created that tackled two common image manipulation techniques that had been previously used in the creation of fake news articles: machine-generated images and image splicing by use of image manipulation software. For the detection of fake imagery, the current state-of-the-art image classification and object detection algorithm, namely Convolutional Neural Networks, was employed with mixed results. On one hand the ensemble model developed for the detection of image splicing, the subjectively easier manipulation technique for human perception, had achieved promising result of 65,05% accuracy and a 0,695 AUROC score on the test images. The ensemble model's lack of better performance can be attributed to two reasons: (1) the comparatively small size of training data employed (2) consequently heterogeneous and high variance in image data that would otherwise be eliminated with the expansion of the training dataset. On the other hand, the ensemble model devised for the detection of images generated using General Adversarial Networks performed exceptionally well on the test images reaching an accuracy of 96,81% and a 0,986 AUROC score.

To put these results into context and to determine the need for such an artifact, 43 human evaluators were tasked to assert the authenticity of images of which some were altered using

splicing and others using a machine-generation technique. The recognition of spliced images by the evaluators reached similar levels (65%) to the performance of the ensemble model, meaning that the model in its current state would not provide any significant value in terms of effectiveness. However, the average evaluator significantly underperformed in comparison to the ensemble model when it came to machine-generated images; these images were capable of deceiving people 43% of the time. As such based on these preliminary results both image manipulation techniques can often times deceive people, stressing the need for detection tools that are capable of aiding the unaware human eyes.

The ensemble model, which is the primary outcome of this research, could be considered as a working prototype for the early detection of fake news articles. There are several proposals which should be implemented in future iterations before the final product is ready for the real-life detection problem. However, even if adequate training is given based on the proposals, the fake image detection algorithm will only detect fake news articles containing image alterations. Given that currently only a subset of fake news articles is assumed to contain falsified visual content, the detection algorithm cannot effectively tackle the fake news problem on its own. Based on prior research in the current news ecosystem the final detection algorithm is proposed to be a detection tool within a larger detection system, relying on content (both textual and visual) and source information of news articles for the early prevention of fake news. To preemptively counter fake news articles from reaching susceptive readers, the early detection system is proposed to be an element of Social Media platforms, in the form of flagging or rating misleading content. Based on previous research, in particular Guess et al. (2018) and Fletcher et al. (2018), Social Media platforms are already heavily employed towards disseminating fake news articles, a trend which is unlikely to stop unless changes are made. Consequently the detection system should be employed as a preventive mean to halt the spreading of fake news at the root of distribution, that is Social Media.

Holistically the prevention of fake news is largely a reactive process rather than proactive, since any change be it technological, platform policy or government intervation cannot stop the “cat-and-mouse” game between fake news providers and protective entities. Fake news providers will constantly diverge their practices given the responses of protective entities e.g., fact-checking practices, changes in search algorithms, policy initiatives. Thus, the best chance platforms and members of the news industry have, is to continuously collaborate with independent researchers in evaluating the scope of the fake news problem and in developing effective means of prevention.

Bibliography

Allcott, H., & Gentzkow, M. (2017). Social Media and Fake News in the 2016 Election. *Journal Of Economic Perspectives*, *31*(2), 211-236. doi: 10.1257/jep.31.2.211

Babakar, M. & Moy, W. (2016). "The State of Automated Factchecking." Full Fact. Retrieved from: https://fullfact.org/media/uploads/full_fact-the_state_of_automated_factchecking_aug_2016.pdf.

Bakir, V., & McStay, A. (2017). Fake News and The Economy of Emotions. *Digital Journalism*, *6*(2), 154-175. doi: 10.1080/21670811.2017.1345645

Bakshy, E., Messing, S., & Adamic, L. (2015). Exposure to ideologically diverse news and opinion on Facebook. *Science*, *348*(6239), 1130-1132. doi: 10.1126/science.aaa1160

Barberá, P., Jost, J. T., Nagler, J., Tucker, J. A., & Bonneau, R. (2015). Tweeting from left to right: Is online political communication more than an echo chamber? *Psychological science*, *26*(10), 1531-1542. doi: 10.1177/0956797615594620

Bengio, Y. (2012). Practical Recommendations for Gradient-Based Training of Deep Architectures.

Bovet, A., & Makse, H. (2019). Influence of fake news in Twitter during the 2016 US presidential election. *Nature Communications*, *10*(1). doi: 10.1038/s41467-018-07761-2

Briggs, R., & Schwabe, G. (2011). *On Expanding the Scope of Design Science in IS Research*. Springer, Berlin, Heidelberg.

Cai, C., Li, L., & Zengi, D. (2017, July). Behavior enhanced deep bot detection in social media. In *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)* (pp. 128-130). IEEE. doi: 10.1109/ISI.2017.8004887

Carlsson, S. (2005). Developing Information Systems Design Knowledge: A Critical Realist Perspective.

Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., & Choo, J. (2017). StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation.

Chollet, F. (2016). Deep Learning with Separable Convolutions.

Chollet, F. (2018). *Deep Learning with Python*. New York: Manning Publications Co.

Ciampaglia, G. L., Shiralkar, P., Rocha, L. M., Bollen, J., Menczer, F., & Flammini, A. (2015). Computational fact checking from knowledge networks. *PloS one*, *10*(6), e0128193. doi: 10.1371/journal.pone.0128193

Dungs, S., Aker, A., Fuhr, N., & Bontcheva, K. (2018). Can Rumour Stance Alone Predict Veracity?. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 3360-3370).

Dunn, T. N. (2016) Revealed: Queen backs Brexit as alleged EU bust-up with ex-Deputy PM emerges. Sun, Retrieved from: *https://www.thesun.co.uk/news/1078504/revealed-queen-backs-brexit-as-alleged-eu-bust-up-with-ex-deputy-pm-emerges/*.

Džeroski, S., Panov, C., & Ženko, B. (2009). Ensemble Methods in Machine Learning.

European Commission. Directorate-General for Communication Networks, Content and Technology. (2018). A multi-dimensional approach to disinformation: Report of the independent High level group on fake news and online disinformation. Publications Office of the European Union.

Figueira, Á., & Oliveira, L. (2017). The current state of fake news: challenges and opportunities. *Procedia Computer Science*, *121*, 817-825. doi: 10.1016/j.procs.2017.11.106

Flaxman, S., Goel, S., & Rao, J. (2016). Filter Bubbles, Echo Chambers, and Online News Consumption. *Public Opinion Quarterly*, *80*(S1), 298-320. doi: 10.1093/poq/nfw006

Fletcher, R., Cornia, A., Graves, L., & Nielsen, R. K. (2018). Measuring the reach of "fake news" and online disinformation in Europe. *Reuters Institute factsheet*, Feb 2018.

Gelfert, A. (2018). Fake News: A Definition. *Informal Logic*, *38*(1), 84-117. doi: 10.22329/il.v38i1.5068

Gil de Zúñiga, H., Weeks, B., & Ardèvol-Abreu, A. (2017). Effects of the news-finds-me perception in communication: Social media use implications for news seeking and learning about politics. *Journal of computer-mediated communication*, *22*(3), 105-123. doi: 10.1111/jcc4.12185

Goodfellow, I. (2016). NIPS 2016 Tutorial: Generative Adversarial Networks.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Bengio, Y. (2014). Generative Adversarial Nets.

Guess, A., Nyhan, B., & Reifler, J. (2018). Selective exposure to misinformation: Evidence from the consumption of fake news during the 2016 US presidential campaign. *European Research Council*, *9*. Retrieved from: http://www.ask-force.org/web/Fundamentalists/Guess-Selective-Exposure-to-Misinformation-Evidence-Presidential-Campaign-2018.pdf

Gulrajani, I., Brain, G., Raffel, C., & Metz, L. (2019). Towards Gan Benchmarks Which Require Generalization.

Hastie, T., Friedman, J., & Tisbshirani, R. (2013). *The Elements of statistical learning* (2nd ed.). New York: Springer.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition.

Hinton Geoffrey. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Hinton, G., Osindero, S., & Teh, Y. (2006). A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, *18*(7), 1527-1554. doi: 10.1162/neco.2006.18.7.1527

Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Adam, H. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. (2018). Densely Connected Convolutional Networks.

Huh, M., Liu, A., Owens, A., & Efros, A. (2018). Fighting Fake News: Image Splice Detection via Learned Self-Consistency.

Ioffe, S., & Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.

Jacobson, S., Myung, E., & Johnson, S. L. (2016). Open media or echo chamber: The use of links in audience discussions on the Facebook pages of partisan news organizations. *Information, Communication & Society*, *19*(7), 875-891. doi: 10.1080/1369118X.2015.1064461

Jin, Z., Cao, J., Zhang, Y., & Luo, J. (2016, March). News verification by exploiting conflicting social viewpoints in microblogs. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*.

Karpathy, A., Li, F. (2018). *CS231n Convolutional Neural Networks for Visual Recognition*. Retrieved from: http://cs231n.github.io/

Karras, T., Laine, S., & Aila, T. (2018). A Style-Based Generator Architecture for Generative Adversarial Networks.

Ketkar, N. (2017). *Deep Learning with Python*. Bangalore: Apress.

Kittler, J. (2000). Multiple classifier systems.

Korshunov, P., & Marcel, S. (2018). DeepFakes: a New Threat to Face Recognition? Assessment and Detection.

Lazer, D. M. J., Baum, M. A.,Benkler, Y.,Berinsky, A. J.,Greenhill, M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., Rothschild D., Schudson M., Sloman, S. A., Sunstein, C. R., Thorson, E. A., Watts, D. J., Zittrain, J. L. The science of fake news: Addressing fake news requires a multidisciplinary effort. *Science*, 359(6380), 1094-1096, doi: 10.1126/science.aao2998

Lee, E., & Tandoc, E. (2017). When News Meets the Audience: How Audience Feedback Online Affects News Production and Consumption. *Human Communication Research*, *43*(4), 436-449. doi: 10.1111/hcre.12123

Levy, N. (2017). The bad news about fake news. *Social Epistemology Review and Reply Collective*, *6*(8), 20-36. Retrieved from: http://wp.me/p1Bfg0-3GV.

Lilleker, D. (2017). Evidence to the Culture, Media and Sport Committee 'Fake news' inquiry presented by the Faculty for Media & Communication, Bournemouth University.

Martens, B., Aguiar, L., Gomez-Herrera, E., & Mueller-Langer, F. (2018). *The digital transformation of news media and the rise of disinformation and fake news-An*

economic perspective. Digital Economy Working Paper 2018-02. Retrieved from: *https://www.researchgate.net/profile/Frank_Mueller-Langer2/publication/325184841_The_Digital_Transformation_of_News_Media_and_the_Rise_of_Disinformation_and_Fake_News/links/5bfe74f0a6fdcc1b8d48700e/The-Digital-Transformation-of-News-Media-and-the-Rise-of-Disinformation-and-Fake-News.pdf*

McCulloch, W., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin Of Mathematical Biology*, *52*(1-2), 99-115. doi: 10.1016/s0092-8240(05)80006-0

Metzger, M., Flanagin, A., & Medders, R. (2010). Social and Heuristic Approaches to Credibility Evaluation Online. *Journal Of Communication*, *60*(3), 413-439. doi: 10.1111/j.1460-2466.2010.01488.x

Miller Robert, & Brewer, J. (2003). The A-Z of Social Research: A Dictionary of Key Social Science Research Concepts

Minsky, M., & Papert, S. A. (1988). *Perceptrons: An introduction to computational geometry*. MIT press.

Mitchell, T. (1997). *Machine learning*. Singapore: McGraw-Hill.

Nataraj, L., Mohammed, T., Manjunath, B., Chandrasekaran, S., Flenner, A., Bappy, J., & Roy-Chowdhury, A. (2019). Detecting GAN generated Fake Images using Co-occurrence Matrices.

Newman, N., & Fletcher, R. (2017). Bias, Bullshit and Lies: Audience Perspectives on Low Trust in the Media. *SSRN Electronic Journal*. doi: 10.2139/ssrn.3173579

Nizza, M. & Lyons, P. J. (2008) In an Iranian Image, a Missile Too Many. New York Times. Retrieved from: *https://thelede.blogs.nytimes.com/2008/07/10/in-an-iranian-image-a-missile-too-many/*.

O'Neil, L. (2019) Doctored video of sinister Mark Zuckerberg puts Facebook to test. The Guardian, Retrieved from: *https://www.theguardian.com/technology/2019/jun/11/deepfake-zuckerberg-instagram-facebook*

Odena, A., & Goodfellow, I. (2018). TensorFuzz: Debugging Neural Networks with Coverage-Guided Fuzzing.

Pattanayak, S. (2017). Introduction to Deep-Learning Concepts and TensorFlow. doi:10.1007/978-1-4842-3096-1_2.

Pérez-Rosas, V., Kleinberg, B., Lefevre, A., & Mihalcea, R. (2017). Automatic detection of fake news. *arXiv preprint arXiv:1708.07104*.

Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised Representation Learning With Deep Convolutional Generative Adversarial Networks.

Ragas, M., Tran, H., & Martin, J. (2013). Media-Induced Or Search-Driven? A study of online agenda-setting effects during the BP oil disaster. *Journalism Studies*, *15*(1), 48-63. doi: 10.1080/1461670x.2013.793509

Rapoza, K. (2017). Can 'fake news' impact the stock market? Forbes, Retrieved from: *https://www.forbes.com/sites/kenrapoza/2017/02/26/can-fake-news-impact-the-stock-market/*.

Real and Fake Face Detection. (2019). *Retrieved from https://www.kaggle.com/ciplab/real-and- fake-face-detection on 2019-09-14*

Reed, R., & Marks, R. (1999). *Neural smithing: Feedforward Artifical Neural Networks*. Cambridge, Mass.: CogNet.

Rochlin, N. (2017). Fake news: belief in post-truth. *Library Hi Tech*, *35*(3), 386-392. doi: 10.1108/lht-03-2017-0062

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386-408. doi: 10.1037/h0042519

Rumelhart, D., Hinton, G., & Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533-536. doi: 10.1038/323533a0

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks.

Sarlin, B. (2018) 'Fake news' went viral in 2016. This expert studied who clicked. NBC News, Retrieved from: *https://www.nbcnews.com/politics/politics-news/fake-news-went-viral-2016-expert-studied-who-clicked-n836581*.

Saunders, M., Lewis, P., & Thornhill, A. (2016). Research methods for business students.

Schwartz, O. (2018) You thought fake news was bad? Deep fakes are where truth goes to die. The Guardian, Retrieved from: *https://www.theguardian.com/technology/2018/nov/12/deep-fakes-fake-news-truth*.

Shao, C., Ciampaglia, G. L., Varol, O., Flammini, A., & Menczer, F. (2017). The spread of fake news by social bots. *arXiv preprint arXiv:1707.07592*, 96-104.

Sharda, R., & Voß, S. (2010). Integrated Series in Information Systems.

Sharma, S., Shanmugasundaram, K., & Ramasamy, S. (2017). FAREC - CNN based efficient face recognition technique using Dlib. Proceedings of 2016 International Conference on Advanced Communication Control and Computing Technologies, ICACCCT 2016 (pp. 192-195). Institute of Electrical and Electronics Engineers Inc.

Shearer, E. (2018) Social media outpaces print newspapers in the U.S. as a news source. *Retrieved from https://www.pewresearch.org/fact-tank/2018/12/10/social-media-outpaces-print-newspapers-in-the-u-s-as-a-news-source/ on 2019-09-14*

Shearer, E., & Gottfried, J. (2016). *News Use Across Social Medial Platforms 2016*. Pew Research Center. Retrieved from: *http://www.journalism.org/2016/05/26/news-use-acrosssocial-media-platforms-2016*.

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake News Detection on Social Media. *ACM SIGKDD Explorations Newsletter*, *19*(1), 22-36. doi: 10.1145/3137597.3137600

Silverman, C. (2016) This analysis shows how viral fake election news stories outperformed real news on Facebook. Buzzfeed News, Retrieved from: *https://www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook*.

Simonyan, K., & Zisserman, A. (2015). Very Deep Convolutional Networks For Large-Scale Image Recognition.

Smith, A., Anderson, M. (2018) Social Media Use in 2018. *Retrieved from https://www.pewinternet.org/2018/03/01/social-media-use-in-2018/ on 2019-09-14*

Spohr, D. (2017). Fake news and ideological polarization. *Business Information Review*, *34*(3), 150-160. doi: 10.1177/0266382117722446

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. (2016). Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning.

Tandoc, E. (2014). Why Web Analytics Click. *Journalism Studies*, *16*(6), 782-799. doi: 10.1080/1461670x.2014.946309

Tandoc, E., Lim, Z., & Ling, R. (2017). Defining "Fake News". *Digital Journalism*, *6*(2), 137-153. doi: 10.1080/21670811.2017.1360143

Tandoc, E., Ling, R., Westlund, O., Duffy, A., Goh, D., & Zheng Wei, L. (2017). Audiences' acts of authentication in the age of fake news: A conceptual framework. *New Media & Society*, *20*(8), 2745-2763. doi: 10.1177/1461444817731756

Thielman, S. (2016) Facebook news selection is in hands of editors not algorithms, documents show. The Guardian, Retrieved from: *https://www.theguardian.com/technology/2016/may/12/facebook-trending-news-leaked-documents-editor-guidelines*

Torrey, L. (2009). Relational Transfer in Reinforcement Learning.

Turcotte, J., York, C., Irving, J., Scholl, R., & Pingree, R. (2015). News Recommendations from Social Media Opinion Leaders: Effects on Media Trust and Information Seeking. *Journal Of Computer-Mediated Communication*, *20*(5), 520-535. doi: 10.1111/jcc4.12127

Vosoughi, S., Roy, D., & Aral, S. (2018). The spread of true and false news online. *Science*, *359*(6380), 1146-1151. doi: 10.1126/science.aap9559

Warren, R. (2016) A fake photoshopped photo of this Sikh guy is going viral again. BuzzfeedNews, Retrieved from: *https://www.buzzfeednews.com/article/rossalynwarren/this-sikh-man-has-been-wrongly-accused-of-a-terror-attack-fo#.rgoEvoBvg*.

Welbers, K., van Atteveldt, W., Kleinnijenhuis, J., Ruigrok, N., & Schaper, J. (2016). News selection criteria in the digital age: Professional norms versus online audience metrics. *Journalism: Theory, Practice & Criticism*, *17*(8), 1037-1053. doi: 10.1177/1464884915595474

Xia, X., Xu, C., & Nan, B. (2017). Inception-v3 for flower classification. 2017 2nd International Conference on Image, Vision and Computing, ICIVC 2017 (pp. 783-787). Institute of Electrical and Electronics Engineers Inc.

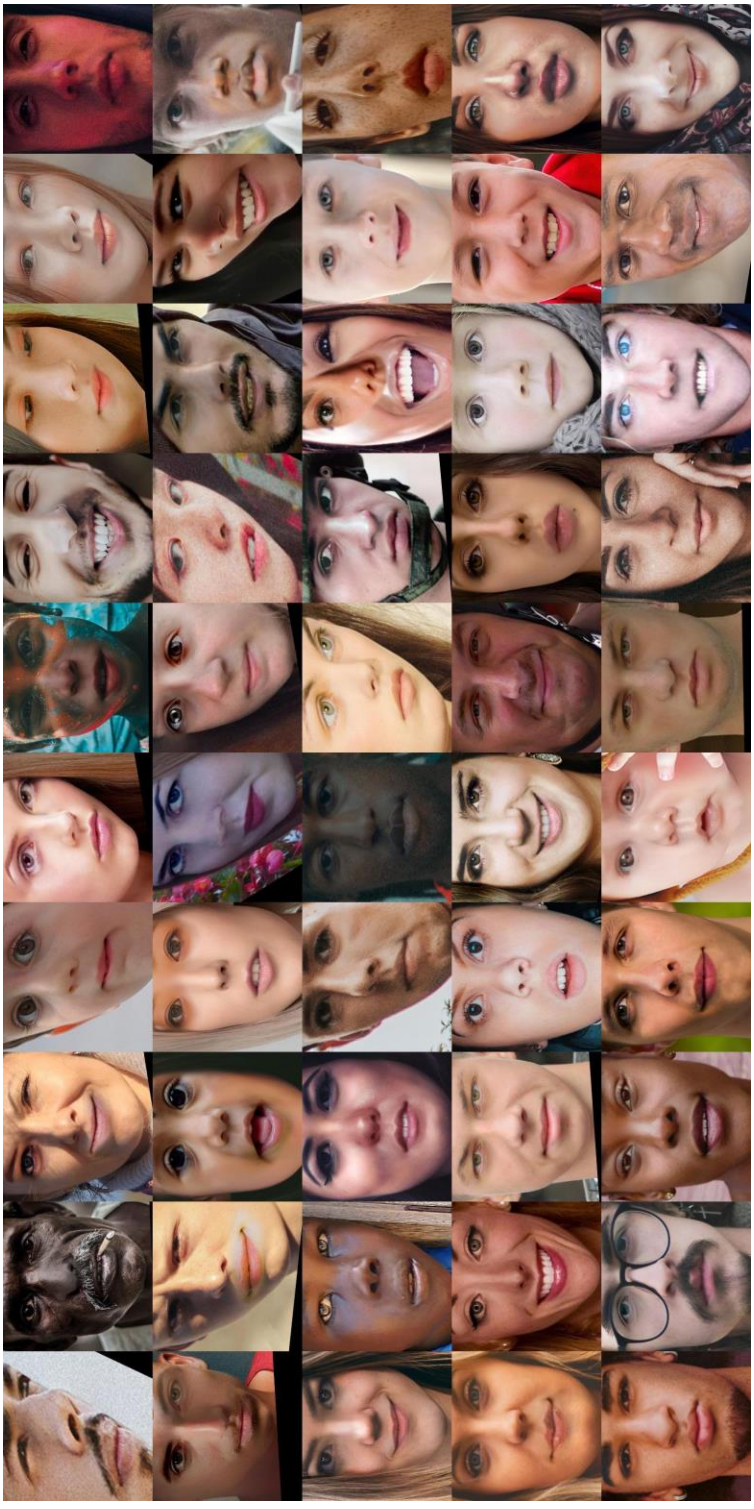
Yaman, M., Subasi, A., & Rattay, F. (2018). Comparison of Random Subspace and Voting Ensemble Machine Learning Methods for Face Recognition. Symmetry, 10(11), 651.

Zhao, Z., Resnick, P., & Mei, Q. (2015, May). Enquiring minds: Early detection of rumors in social media from enquiry posts. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 1395-1405). International World Wide Web Conferences Steering Committee. doi: 10.1145/2736277.2741637.

Zoph, B., Vasudevan, V., Shlens, J., & Le, Q. (2017). Learning Transferable Architectures for Scalable Image Recognition.

Appendices

Appendix A



Appendix A: Continuum representation of confidence on fake images of the Photoshop Dataset. Each row representing a 20th percentile of the confidence. Top row: False Positive with very high confidence (0.8-1); Bottom row: True Positive with very high confidence (0-0.2); Each row contains 10 images

Appendix B

Appendix B: Continuum representation of confidence on fake images of the GAN Dataset. Each row representing a 20th percentile of the confidence. Top row: False Positive with very high confidence (0.8-1); Bottom row: True Positive with very high confidence (0-0.2); Each row contains 10 images

Appendix C

Appendix C is an attached USB Drive containing the datasets used for training and evaluation as well as the codebase used during training and evaluation. It contains six folders:

Image Data

Contains the raw images, extracted images using the dlib library and the images used in the survey

Code Repository

Contains the written code for image preprocessing, extraction, classifier training, evaluation and the code used for visualization.

Final Models

Contains the base model architectures and the trained weights files for the models. These can be used to retrieve our base models

Test dataset evaluations

Includes .csv and .xlsx files which contain the FaRe-GAN and FaRe-PS predictions on each test images and the aggregated results.

Survey data

Contains the raw survey data collected from the evaluators and the aggregated results, which were used in the tables within the thesis paper. Additionally, the FaRe-GAN and FaRe-PS predictions on the 100 images that were used in the survey.

Continuum grid images

Contains visualizations of the misclassifications by the FaRe-GAN and the FaRe-PS models in different formats from 6x9 to 64x5.