D6.1

First generation of Human- and Society-centered AI algorithms

Project Title	AI4Media - A European Excellence Centre for Media, Society and Democracy
Contract No.	951911
Instrument	Research and Innovation Action
Thematic Priority	H2020-EU.2.1.1. - INDUSTRIAL LEADERSHIP - Leadership in enabling and industrial technologies - Information and Communication Technologies (ICT) / ICT-48-2020 - Towards a vibrant European network of AI excellence centres
Start of Project	1 September 2020
Duration	48 months

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951911

info@ai4media.eu www.ai4media.eu

Deliverable title	First generation of Human- and Society-centered AI algorithms
Deliverable number	D6.1
Deliverable version	1.0
Previous version(s)	-
Contractual date of delivery	December 31, 2021
Actual date of delivery	December 21, 2021
Deliverable filename	AI4Media_D6_1.pdf
Nature of deliverable	Report
Dissemination level	Public
Number of pages	109
Work Package	WP6
Task(s)	T6.1-T6.7
Partner responsible	FhG-IDMT
Editor	Thomas Köllmer
Officer	Evangelia Markidou

Abstract	This deliverable presents the initial results of WP6 of AI4Media: Human and Society-centred AI, spanning the first 16 months of the project. The document presents the outcomes grouped by the work package's tasks, highlighting each partners' contribution to AI4Media. The individual tasks are "Policy recommendations for content moderation", "Manipulation and synthetic content detection in multimedia", "Privacy-enhanced recommendation", "AI for Healthier Political Debate", "Detection of perceptions of hyper-local news", "Measuring and Predicting User Perception of Social Media" and "Real-life effects of private content sharing".
Keywords	content manipulation, content moderation, content synthesis, deepfake, hyper-local news, manipulation detection, online political debate, policy recommendations, privacy-aware recommendation, private content sharing, synthetic content detection, user perception measurement

www.ai4media.eu

info@ai4media.eu

Copyright

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951911

Authors & Contributors

Name	Organization
Ioanna Koroni	AUTH
Dmitry Gnatyshak	BSC
Julien Tourille	CEA
Adrian Popescu	CEA
Georgia Pantalona	CERTH
Symeon Papadopoulos	CERTH
Christoforos Papastergiopoulos	CERTH
Luca Cuccovillo	FHG-IDMT
Thomas Köllmer	FHG-IDMT
Daniel Gatica-Perez	IDIAP
David Alonso del Barrio	IDIAP
Lidia Dutkiewicz	KUL
Roberto Caldelli	MICC-UNIFI
Alberto Del Bimbo	MICC-UNIFI
Leonardo Galteri	MICC-UNIFI
Marco Formentini	UNITN
Nicu Sebe	UNITN
Mihai Gabriel Constantin	UPB
Cristian Stanciu	UPB
Georgios Zoumpourlis	QMUL

Peer Reviews

Name	Organization
Danae Tsabouraki	ATC
Killian Levacher	IBM

Revision History

Version	Date	Reviewer	Modifications
0.1	19.11.2021	Thomas Köllmer	First draft with contributions from all partners
0.2	1.12.2021	Thomas Köllmer	Second draft with contributions from all partners
0.3	12.12.2021	Killian Levacher	Updated version with review from Killian Levacher
0.4	17.12.2021	Danae Tsabouraki	Updated version with review from Danae Tsabouraki
1.0	21.12.2021	Thomas Köllmer	Final version.

The information and views set out in this report are those of the author(s) and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf.

Table of Abbreviations and Acronyms

Abbreviation	Meaning
AI	Artificial Intelligence
AUC	Area Under the Curve
CNN	Convolutional Neural Network
DMCA	Digital Millennium Copyright Act
DnCNN	Denoising CNN
DNN	Deep Neural Network
EEG	Electroencephalogram
FG	Feature Generation
FoR	Fake or Real
GAN	Generative Adversarial Network
GDPR	General Data Protection Regulation
GIQA	Generated Image Quality Assessment
MFCC	Mel-Frequency Cepstral Coefficients
MLP	MultiLayer Perceptron
OSN	Online Social Networks
SGD	Stochastic Gradient Descent
SNR	Signal to Noise Ratio
STFT	Short-Time Fourier Transform
ToS	Terms of Service
TTS	Text-to-Speech
V	Vocoding

Contents

1 Executive summary 9

2 Introduction 10
2.1 Structure of this document . . . 10

3 Policy recommendations for content moderation (T6.1) 11
3.1 What is content moderation . . . 11
3.2 The algorithmic content moderation . . . 12
3.3 Technical limitations of automation in content moderation . . . 13
3.4 Algorithmic content moderation challenges for freedom of expression and other fundamental rights . . . 14
3.5 EU regulatory framework on online content moderation . . . 16
3.5.1 Transparency obligations for the use of AI in content moderation . . . 17
3.6 Conclusion and future research direction . . . 19

4 Manipulation and synthetic content detection in multimedia (T6.2) 21
4.1 Vision-based DeepFake detection and generation . . . 22
4.1.1 DeepFake Video Detection with Facial Features and Long-Short Term Memory Deep Networks . . . 22
4.1.2 DeepFake video detection by means of Optical Flow based CNN . . . 25
4.1.3 Detection of synthetically generated images . . . 30
4.1.4 Layout-to-Image Translation with Double Pooling Generative Adversarial Networks . . . 35
4.2 Audio-based DeepFake detection . . . 41
4.2.1 Synthetic speech generation . . . 41
4.2.2 Audio DeepFake detector . . . 43
4.2.3 Microphone classification in noisy environments . . . 46
4.3 Text-based DeepFake detection . . . 49
4.3.1 Tweet Generation and DeepFake Detection . . . 49

5 Hybrid, privacy-enhanced recommendation (T6.3) 56

6 AI for Healthier Political Debate (T6.4) 58
6.1 Neural knowledge transfer for improved sentiment analysis in texts with figurative language . . . 58
6.2 Public opinion monitoring via semantic analysis of tweets . . . 61
6.3 Political tweets in the Greek language . . . 63
6.4 Measuring healthiness of online discussions on Twitter using the temporal dynamics of attention data . . . 64

7 Perception of hyper-local news (T6.5) 68
7.1 Classifying covid-19-related false information in online news articles . . . 68
7.2 Building a corpus of local news about covid-19 vaccination across European countries 69
7.3 Exploring other health information sources: the case of online video . . . 70

8 Measuring and Predicting User Perception of Social Media (T6.6) 74
8.1 Benchmarking and Predicting Media Interestingness in Images and Videos 74
8.2 Predicting Video Memorability 77
8.3 DeepFusion Ensembling Systems 79
8.4 Pairwise Ranking Network for Affect Recognition 84
8.5 Estimating continuous affect with label uncertainty 87

9 Real-life effects of private content sharing (T6.7) 91

10 Summary and Conclusion 96

1 Executive summary

This document shows the current status of WP6: Human and Society-centred, presenting the results achieved so far in project month 16. The goal of the work package is to leverage AI technologies to serve citizens and societies and counter negative impacts that a naive use of AI technologies can have. For AI4Media, this work package has a central role, linking the outcomes of the more algorithm-centric work packages WP3-WP5 and the use cases from WP8.

This is an ambitious goal, also covering a lot of different topics, as reflected by the task structure. Still, WP6 made good process so far, with a lot of excellent contributions by the partners within the individual tasks, but also starting to cooperate across tasks and across work packages. The results presented here give a good start and foundation for the upcoming years of the project and help to build AI components within AI4Media to benefit citizens and societies.

AI technologies cannot be deployed without a legal context. But today, regulations seem to be a step behind the technological advances. A field where this is particularly relevant is content moderation, where technical filtering applications are in a potential conflict with human rights, especially the freedom of expression. To handle this in AI4Media we research current and possible future policies for content moderation in Task 6.1.

DeepFakes – the algorithmic generation and manipulation of content using AI – have a high potential of destroying trust of citizens when they consume media. Therefore, the detection of such synthetic content is of high importance to the project. In this work package, we performed research on detecting fakes in the visual domain, but also detecting fakes in audio and textual data as part of Task 6.2.

Social media and content platforms without recommender systems are inconceivable today. But it showed that recommender systems are the main root of societal problems such as filter bubbles. Also, the current state of the art of recommender systems relies on having a lot of user data with numerous privacy implications. Task 6.3 aims at building a recommender system for AI4Media without such a negative societal impact, handling a specific use case from WP8, namely the Smart News Assistant, and integrating analysis components from the other work packages.

While there are arguably now more online discussions than ever before, certain dynamics led to a lot of poisoned discussion on social media, shit storms, hate speech and the like. Task 6.4 aims at investigating those effects, studying automatic sentiment analysis and opinion mining, as well as creating general measures for discussion quality and public opinions on Twitter. Moreover, research is done on Greek tweets to shine a light on more than just the English-speaking part of the Internet. This is particularly relevant, as online discussions are not globally uniform, but different from country to country and community to community. Task 6.5 researches the perception of local news, in the context of the Covid-19 pandemic. Also we are building a corpus of local news from different European countries on this topic for later analysis.

Task 6.6 turns directly on social networks and how the content there is perceived by a human in terms of memorability and interestingness. In other words, we research on how attention grabbing a certain piece of media is, analysing it across modalities. Finally, Task 6.7 investigates how to use AI to assess the privacy impact of sharing a certain object on social media.

This deliverable alone presents the work of at least 14 publications and relevant software that show the research effort in the first 16 months of the project. In the remainder of the project, this work will be continued with a focus on integrating the outcomes of WP6 with the other work packages and the use cases even more.

2 Introduction

This deliverable presents the results achieved in WP6: Human and society centred AI, during the first 16 months of the AI4Media project. An important goal of this work package is to provide countermeasures against the societal problems that arose due to the proliferation of social networks and the negative effects, unrestricted use of AI tools can have on our society. In the *description of action*, the goals of this task are formulated as follows:

> This WP will develop methods to put AI technologies to the service of citizens and societies. It links EU AI vision elements from WP2 and algorithmic inputs from WP3-WP5 to AI4Media use cases developed in WP8. WP6 activities are structured around AI technologies which are both challenging research-wise and have significant impacts on users' real-life. In particular, methods and tools will be proposed to better understand the factors underpinning political information production and consumption. First, appropriate policies for content moderation in the EU will be examined. Second, as automatic manipulation of multimedia documents has the potential to lead to large-scale misinformation, we will build novel AI algorithms to counter the malicious use of such technologies. Third, new generation recommenders whose objective goes beyond mere personalisation of online experience will be introduced. Fourth, the interaction of citizens with political news will be studied both at European/national and hyper-local levels in order to contribute to a healthier democratic debate. Finally, users' perception of social media and their effects in real life will be investigated.

Ironically, this description was written before the Covid-19 pandemic. While it was already clear two years ago that the way political debates unfold differs dramatically whether a social network is involved or not, the discussions on Covid-19 countermeasures, provenance of the virus or advantages and downsides of the available vaccinations underline boldly the importance of this work package. Events like the attack on the US Capitol in January 2021 and the role of social media and the news also show the need for technical solutions to some of the problems.

Also, the capabilities of generative neural networks increased tremendously, as anticipated in the proposal. So it is even more likely that a deceptively real deep fake will have a measurable societal impact anytime soon and we are in need of technologies that can detect such content.

This deliverable presents the outcomes of the research activities performed in the context of WP6: *Human- and Society-centred AI* during the first 16 months of the project. All referenced use cases are elaborated in great detail in *D8.1: Use Cases Definitions and Requirements.*

2.1 Structure of this document

The presented results are grouped by the tasks of the work package and presented individually in the following sections and are concluded with a summary.

- T6.1 - Policy recommendations for content moderation (page 11)
- T6.2 - Manipulation and synthetic content detection in multimedia (page 21)
- T6.3 - Hybrid, privacy-enhanced recommendation (page 56)
- T6.4 - AI for Healthier Political Debate (page 58)
- T6.5 - Perception of hyper-local news (page 68)
- T6.5 - Measuring and Predicting User Perception of Social Media (page 74)
- T6.6 - Real-life effects of private content sharing (page 91)

3 Policy recommendations for content moderation (T6.1)

Contributing partners: KUL

The main focus of Task 6.1. is to investigate aspects of future regulation of content moderation. The main issue to resolve in the coming years consists in determining who should decide which content should be removed, for which reasons, when and how. What should be the model for content moderation: can the problem be addressed though self-regulation (such as codes of practice, codes of conducts), or is there a need for a hard-law EU regulatory instrument? Can a private instrument, such as Facebook's Oversight Board provide an answer? How should the new content regulation approach be designed to respect fundamental rights such as a freedom of expression without limiting the open public debate? How do we ensure that legitimate, lawful content is not deleted, and that the freedom of expression is not violated? How do users know what gets deleted, and whether what gets deleted violates laws or not? In this task, we will conduct research to provide answers and policy recommendations to these questions.

This section offers an introduction into the main concepts such as content moderation and the use of automated tools in content moderation. Then, it offers a critical assessment of the technical limitations of the algorithmic content moderation and points out the main risks for the fundamental human rights, such as freedom of expression. Finally, it introduces the main elements of the EU regulatory framework applicable to content moderation.

3.1 What is content moderation

Internet intermediaries, typically, are private entities that provide commercial and technical infrastructure which allows information to be exchanged. Because of their enabling role and technical capabilities to affect directly and indirectly the behaviour and content of their users, they hold a powerful position, and are sometimes referred to as *information gatekeepers*. They can eliminate access to a particular service, remove content, amplify or downgrade information they choose to present [1]. In a broad sense, content moderation may therefore be understood as "governance mechanisms that structure participation in a community to facilitate cooperation and prevent abuse" [2]. Content moderation happens at many levels. It can take place before content is actually published on the website (ex ante moderation), or after content is published (ex post moderation). Moreover, moderators can passively assess content only after others flag the content to their attention, or they can proactively seek out published content for removal. Additionally, content moderation decisions can be made by automated means (using Artificial Intelligence (AI)) or manually made by human content moderators [3]. The focus of this section is on the use of automated means. In content moderation, automation can be used in different phases of content moderation processes: proactive detection of potentially problematic content, the automated evaluation, or the enforcement of a decision to remove, demonetize, amplify, or prioritize content.

To better illustrate the scale of content moderation on major social media platforms, only in the third quarter of 2021, Facebook *took action* on 34.7 million pieces of *adult nudity and sexual activity content*, 9.2 million pieces of *bullying and harassment content*, 20.9 million pieces of *child sexual exploitation content*, 22.3 million pieces of *hate speech content* and 13.6 million pieces of *violence and incitement content.* It also took action on 1.8 billion of fake accounts [4]. Between April and June 2021, YouTube removed 4.1 million channels and 1 billion comments [5]. Between July and December 2020, Twitter *actioned* on 3.5 million accounts, suspended 1 million accounts and removed 4.5 million pieces of content [6]. These numbers only represent cases where platforms acted; the overall number of decisions considered, including those where no action was taken, is of course much higher.

Due to the massive amount of content uploaded on social media platforms every second, it is clear that platforms must, in some or another form, moderate the content. The scale at which these platforms operate means mistakes in enforcing any rule are inevitable: it will always be possible to find examples of both false positives and false negatives [7]. The challenge for platforms, is exactly when, how, and why to intervene [8].

Some content takedowns are required by law, while others are performed voluntarily by platforms. Legally required removals are shaped by intermediary liability laws, which tell platforms what responsibility they have for illegal content posted by their users [9]. Platforms operating under legal frameworks like the US Digital Millennium Copyright Act (DMCA)[1] or the EU's eCommerce Directive[2] typically work under *notice-and-action* systems. *Notice and action* is an umbrella term for a range of mechanisms designed to eliminate illegal content from the Internet. According to the European Commission, "[t]he notice and action procedures are those followed by the intermediary internet providers for the purpose of combating illegal content upon receipt of notification. The intermediary may, for example, take down illegal content, block it, or request that it be voluntarily taken down by the persons who posted it online". Platforms' voluntary content removals are based on their own set of rules: Community Standards and Terms of Service (ToS), which often include platform operators' own moral beliefs or social norms.

Moreover, platforms moderate content which belongs to a wide range of categories, including terrorism, graphic violence, toxic speech (hate speech, harassment and bullying), sexual content, child abuse and spam/fake account detection. Clearly, these types of content are fundamentally different, not just in terms of their illegality, but also their characteristics and the gravity of their consequences. It is crucial to recognise that different types of content moderation are fundamentally different and that there is no one size fits all solution which may be appropriate in every case. Illegal or potentially problematic content ranges from content that is illegal everywhere to content that is legal but potentially harmful (such as disinformation).

3.2 The algorithmic content moderation

Enormous amounts of content are uploaded and circulated on the Internet every day, far outpacing any intermediary's ability to have humans analyse content before it is uploaded. Many platforms have therefore turned to automated processes to assist in the detection and analysis of illegal or problematic content, including disinformation, hate speech, and terrorist propaganda [10]. Gorwa et al. define algorithmic (commercial) content moderation as "systems that classify user-generated content based on either matching or prediction, leading to a decision and governance outcome (e.g. removal, geoblocking, account takedown)" [9]. Algorithmic content moderation involves a range of techniques from statistics and computer science. There are two main systems used in algorithmic content moderation. First, those that aim to match a newly uploaded piece of content against an existing database. To illustrate, the Global Internet Forum to Counter Terrorism (GIFCT), a hash-sharing database led by Google, Facebook, Twitter and Microsoft, plays a significant role in fighting extremism online by removing content it qualifies as *terrorism related content* under its own terms of service. The technical limitations of hash-sharing technology and the GIFCT database were clearly demonstrated in the Christchurch shooting incident in 2019. On 15 March 2019, a terrorist live streamed on Facebook his attack on a mosque in Christchurch, New Zealand in which he killed more than 50 people. The live video of the shooting went viral around the world and was able to play for 17 minutes before it was taken down. Including the views during the live broadcast, the video was viewed about 4,000 times in total before being removed from Facebook. The video was taken down not before hundreds of thousands of versions were made and

[1] https://www.copyright.gov/legislation/dmca.pdf
[2] https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32000L0031&from=EN

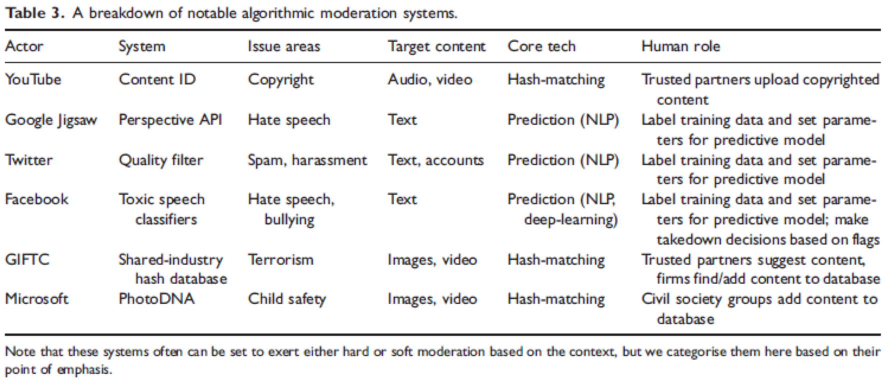


Table 3. A breakdown of notable algorithmic moderation systems.

Actor	System	Issue areas	Target content	Core tech	Human role
YouTube	Content ID	Copyright	Audio, video	Hash-matching	Trusted partners upload copyrighted content
Google Jigsaw	Perspective API	Hate speech	Text	Prediction (NLP)	Label training data and set parameters for predictive model
Twitter	Quality filter	Spam, harassment	Text, accounts	Prediction (NLP)	Label training data and set parameters for predictive model
Facebook	Toxic speech classifiers	Hate speech, bullying	Text	Prediction (NLP, deep-learning)	Label training data and set parameters for predictive model; make takedown decisions based on flags
GIFTC	Shared-industry hash database	Terrorism	Images, video	Hash-matching	Trusted partners suggest content, firms find/add content to database
Microsoft	PhotoDNA	Child safety	Images, video	Hash-matching	Civil society groups add content to database

Note that these systems often can be set to exert either hard or soft moderation based on the context, but we categorise them here based on their point of emphasis.

Figure 1. Algorithmic content moderation systems (taken from [9]).

re-uploaded to Facebook, YouTube and Twitter [9]. Hash-sharing efforts failed mainly because initial images did not match closely enough to any images already in the database. There was not enough similar pre-existing content in the database to allow the machine learning system to match mass shooting-related content [11].

The second category includes systems that aim to classify new content into one of a number of categories [9]. This category is even more problematic due to the lack of contextualization as explained below. Figure 1 illustrates a breakdown of notable algorithmic moderation systems [9].

3.3 Technical limitations of automation in content moderation

Llansó et al. point to the following technical limitations of automation in content moderation [10]. First, the importance of context. Whether a particular post amounts to a violation of law or a platforms' Community Standards or Terms of Service often depends on context that the machine learning system does not recognise. A study on the use of AI tools in hate speech detection points out that these tools are not yet able to understand context, irony or satire [12]. The best-known example of a lack of contextual differentiation by an online platform's content moderation decision is Facebook's removal of the iconic *napalm girl* 1972 photo, which depicts a young nude girl running from a napalm attack during the Vietnam War [13]. Facebook removed the photo as it breached their Community Standards stating that "while we recognise that this photo is iconic, it's difficult to create a distinction between allowing a photograph of a nude child in one instance and not others" [14]. Facebook later reversed its decision and re-instated the photo. Whilst many users would agree that child nudity should be removed from online platforms, this example highlights the importance of context when moderating online content.

Moreover, lack of contextual interpretation of the terrorist content risks that legal uses of terrorist material (such as for educational, artistic, journalistic or research purposes, or for awareness raising purposes against terrorist activity) will be deleted. This has happened to the Syrian Archive, a non-profit organization documenting war crimes committed by terrorist organizations. Its content was repeatedly removed from online platforms, including YouTube, for being *extremist* content and thereby violating platforms' Community Standards and ToS. As a result, the removals lead to widespread and sometimes permanent losses of what might be crucial evidence of war crimes for the International Criminal Court (ICC) or other law enforcement authorities [15].

Second, there is a lack of representative, well-annotated datasets to use for machine learning training. Many tools are trained on labelled datasets that are already publicly available. However, if these datasets do not include examples of speech in different languages and dialects, the resulting tools will not be equipped to analyse these groups' communication. According to the recent Facebook Files, in India, Facebook's single biggest market by audience size, with more than 400 million users, the company's systems were falling short in their effort to crack down on hate speech [16]. The AI models lacked classifiers in the local languages that need to be trained to detect and remove content such as hate speech. The company added Hindi hate speech classifiers in 2018, and Bengali hate speech classifiers in 2020. Those two languages are among India's most popular, spoken collectively by more than 600 million people, according to the country's most recent census in 2011 [16]. The lack of Hindi and Bengali classifiers means that the hate speech content was never detected or flagged before the changes made in 2018 and 2020.

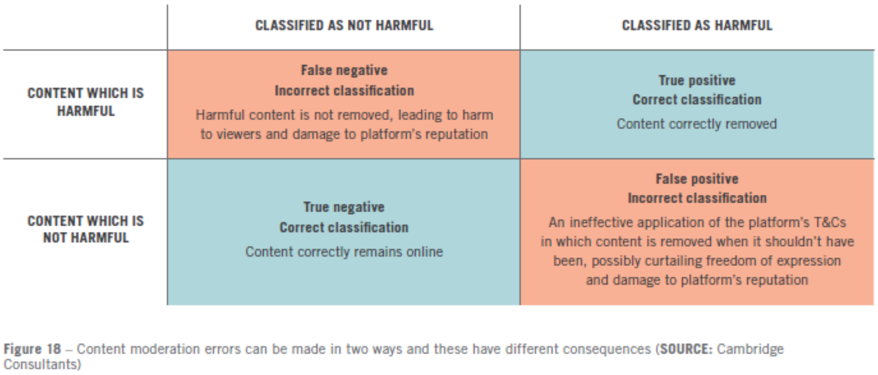
The problem with quality and representation of the training data, especially those in publicly available datasets and databases, is well recognized in the academic literature. As mentioned by Raji and others, "privacy and consent violations in the dataset curation process often disproportionately affect members of marginalized communities. Benchmark dataset curation frequently involves supplementing or highlighting data from a specific population that is underrepresented in previous dataset"[17]. There are a number of studies showing that in the publicly available datasets certain groups are highly underrepresented. The problem is even more visible when it comes to intersectional identities [18]. To this end, it is likely that using such data could lead to algorithmic results being biased and discriminatory.

Moreover, the process of labelling a dataset for supervised learning typically requires the involvement of multiple human reviewers to evaluate examples and select the appropriate label, or to evaluate an automatically applied label. What constitutes *hate speech* or *disinformation* is however a socio-political matter and varies across countries and jurisdictions. The humans applying the label often do not agree among themselves what content merits the label of, for example, *hate speech* or *spam*.

3.4 Algorithmic content moderation challenges for freedom of expression and other fundamental rights

Beyond the technical limitations of automated content moderation, the use of automation in content moderation systems raises challenges for freedom of expression and other fundamental rights, which is the main focus of Task 6.1. Importantly, the right to freedom of expression is enshrined in Article 10 of the European Convention of Human Rights (ECHR)[3]. It includes the right to freely express opinions, views, ideas and to seek, receive and impart information regardless of frontiers. Freedom of expression is applicable not only to *information* or *ideas* that are favourably received or regarded as inoffensive or as a matter of indifference, but also to those that offend, shock or disturb. Users have therefore the right to receive and impart information on the Internet, in particular to create, re-use and distribute content using the Internet. The right to freedom of expression in Europe has a broad scope of application. It is not limited to citizens, or natural persons only. The right protects any expression regardless of its content, its form (any word, picture, image or action to express an idea, etc.), its speaker, or the type of medium used. There is, however, expression that does not qualify for protection under Article 10 of the ECHR such as an incitement to violence, hate speech directed towards different and speech promoting the Nazi ideology and denying the Holocaust. Moreover, the right to freedom of expression is not an absolute right. The exercise of the right to freedom of expression may be subject to formalities, conditions, restrictions or penalties under three conditions. In particular, the restriction must be (1) prescribed by law, (2) introduced for protection of a legitimate aim (e.g. protection of the rights of others) and (3) necessary in a

[3] https://www.echr.coe.int/documents/convention_eng.pdf



	CLASSIFIED AS NOT HARMFUL	CLASSIFIED AS HARMFUL
CONTENT WHICH IS HARMFUL	False negative Incorrect classification Harmful content is not removed, leading to harm to viewers and damage to platform's reputation	True positive Correct classification Content correctly removed
CONTENT WHICH IS NOT HARMFUL	True negative Correct classification Content correctly remains online	False positive Incorrect classification An ineffective application of the platform's T&Cs in which content is removed when it shouldn't have been, possibly curtailing freedom of expression and damage to platform's reputation

Figure 18 – Content moderation errors can be made in two ways and these have different consequences (**SOURCE:** Cambridge Consultants)

Figure 2. Content moderation errors and their consequences. The source of this figure is Ofcom's report on the "Use of AI in online content moderation" [4].

democratic society (proportionate). The rules defining the conditions for lawful interference with expression are addressed to the States and not to private entities [1].

There is a growing body of literature on the human rights implications of the use of automation by online platforms: Special Rapporteur on the promotion and protection of the right to freedom of opinion and expression, David Kayne has issued a Report on AI technologies and implications for freedom of expression and the information environment[19]. The Council of Europe has provided several reports, studies, and recommendations that touch on the topic and published in May 2021 its Guidance Note on Content Moderation[20]. The use of automated means by online platforms has also been subject addressed by the Court of Justice of the European Union (CJEU). In SABAM v. Netlog, the Court held that a filtering system could "undermine freedom of information since that system might not distinguish adequately between lawful and unlawful content, with the result that its introduction could lead to the blocking of lawful communications"[21].

There are a number of recognized issues with the application of algorithmic systems and automation for these purposes. First, the use of algorithmic systems for detecting particular types of speech and activity will always have so-called false positives (something is wrongly classified as objectionable) and negatives (the automated tool misses something that should have been classified as objectionable). From a freedom of expression perspective, false positives risk infringing individuals' right to freedom of expression. Online platforms operate under circumstances in which the cost of overmoderation is low, which makes them set up their content moderation systems to by default remove online content or suspend the accounts [19]. False negatives, on the other hand, can result in a failure to address hate speech, and may create a chilling effect on some individuals' and groups' willingness to participate online [10]. Figure 2 illustrates content moderation errors and their consequences.

Second, content moderation requires the processing of a range of personal data. A range of personal and non-personal data must be stored by the company, such as the username of the individual, the name of the complainant, the justification for the removal of the content, dates and times of uploads and removals and so on. Furthermore, the processing of such data may include the processing of special categories of data such as in relation to political opinions, trade union memberships, religious or other beliefs. Such data may only be processed under the General Data Protection Regulation (GDPR) and Convention 108(+), if appropriate safeguards exist in law. Moreover, algorithmic content moderation systems will typically rely on the large-scale processing of user data. This may also involve profiling of users, which is again problematic from a fundamental rights perspective. In this way, the growing reliance on algorithmic systems further encourages the collection and processing of personal data, which pose additional risks to the rights to privacy and freedom of expression [10].

Third, algorithmic systems have the potential to perform badly on data related to underrepresented groups, including racial and ethnic minorities, non-dominant languages, and/or political leanings. This can result in serious risks to freedom of expression for communities and individuals, including illegitimate silencing of their expression and failure to address harms to their communities. As a result, vulnerable groups are the most likely to be disadvantaged by AI content moderation systems [10].

Forth, there is a growing need for redress and accountability for online platforms for making determinations about speech, especially given the enormous scale of speech that is being evaluated. When content is removed, it is important that transparency measures make clear the specific reasons why the content was removed. The right to effective remedy, including complaint, review, and appeal procedure for people whose content has been unjustly removed must be ensured.

3.5 EU regulatory framework on online content moderation

The EU regulatory framework on content moderation is increasingly complex and has been differentiated over the years according to the category of the online platform, the type of content and the nature of the legal instrument (hard-law, soft-law, or self-regulation).

The main elements of the EU regulatory framework include first, horizontal rules applicable to all categories of online platforms and to all types of content, i.e. the e-Commerce Directive. The goal of that directive is to allow borderless access to digital services across the EU and to harmonise the core aspects for such services, including information requirements and online advertising rules. The Directive applies to any kind of illegal or infringing content. It sets out the framework for the liability regime of intermediary services – categorised as *mere conduits*, *caching services*, and *hosting services* – for third party content. Under Article 14 of the e-Commerce Directive, hosting providers can benefit from a liability exemption provided they act expeditiously to remove or disable access to information upon obtaining knowledge about its illegal character. The provider of a hosting service can obtain knowledge about the illicit character of hosted content through his own activities or he could be notified by a private individual to take down the content in question (notice and takedown). As a result, it becomes the provider's task to assess whether the complaint is justified and to make a decision about its illegal or infringing character of the content. The provider can either leave the content on its platform and risk liability for it, or relieve himself of the problem altogether by simply removing the content [1].

On 15 December 2020, the European Commission has proposed a comprehensive set of new rules for digital services, including social media, online market places, and other online platforms that operate in the European Union: the Digital Services Act (DSA) and the Digital Markets Act. The DSA proposal maintains the liability rules for providers of intermediary services set out in the e-Commerce Directive – by now established as a foundation of the online sphere. However, other rules of the eCommerce Directive adopted 20 years ago will be revised.

The main aims of the new rules are to:

- establish a horizontal framework for regulatory oversight, accountability and transparency of the online space;
- improve the mechanisms for the removal of illegal content and for the effective protection of users' fundamental rights online, including the freedom of speech;
- propose rules to ensure greater accountability on how platforms moderate content, on advertising and on algorithmic processes;
- provide users with possibilities to challenge the platforms' decisions to remove or label content;

- impose new obligations on very large online platforms (VLOPS) to assess the risks their systems pose and to develop appropriate risk management tools to protect the integrity of their services against the use of manipulative techniques;
- clarify responsibilities and accountability for online platforms and to provide new powers to scrutinize how platforms work, including by facilitating access by researchers to key platform data.

The DSA proposal considers the impact of the use of AI based tools used in online media. The preamble of the proposal underlines how algorithmic systems shape information flows online (e.g. via content prioritization, advertisement display and targeting or content moderation). It further points to the need expressed by civil society and academics for algorithmic accountability and transparency audits, especially about how information is prioritized and targeted to users.

3.5.1 Transparency obligations for the use of AI in content moderation

According to the DSA proposal, providers of intermediary services must include in their terms and conditions, in a clear and ambiguous language, information on any policies, procedures, measures and tools used for the purpose of content moderation, including "algorithmic decision-making" and human review. Providers of hosting services shall put mechanisms in place to allow any individual or entity to notify them of the presence on their service of specific items of information that the individual or entity considers to be illegal content. Online platforms must also put in place an internal complaint-handling system for managing the complaints against a decision taken against information provided/uploaded by a recipient of their services. The decision on the complaint must not be solely taken based on automated means. The online platforms also have additional obligations when it comes to transparency reporting as they must include in their yearly report on content moderation, any use made of automatic means for the purpose of content moderation, including a specification of the precise purposes, indicators of the accuracy of the automated means in fulfilling those purposes and any safeguards applied. The detailed assessment of the DSA rules will be provided in deliverable D6.2.

Second, there are some stricter rules applicable to Video-Sharing Platforms (VSPs) and to certain types of illegal content online, i.e. the revised Audio-visual Media Service Directive (AVMSD).

Third, there are vertical rules applicable to the four types of illegal content, which are illegal under EU law (i.e. terrorist content, child sexual abuse material, racist and xenophobic hate speech and violations of Intellectual Property). In particular, recently adopted regulation on preventing the dissemination of terrorist content online has been subject of many controversies. It requires terrorist content to be removed within one hour from the receipt of the removal order and imposes financial penalties for non-compliance. The one-hour response deadline has been critised for being very difficult to meet in practice and for creating incentives for over-removal of content. Multiple NGOs have also underlined that such removal orders "must be met within this short time period regardless of any legitimate objections platforms or their users may have to the removal of the content specified, and the damage to freedom of expression and access to information may already be irreversible by the time any future appeal process is complete"[22]. In practice, hosting service providers would rather prefer to delete more content, faster, e.g. by installing upload filters to systematic monitoring the entirety of the users' content, than face financial penalties. This risks negatively affecting fundamental rights, in particular the right to freedom of expression.

Similarly, the Copyright in Digital Single Market Directive (CDSM) established a new liability regime for online content-sharing platforms. Article 17, the most controversial and hotly debated provision of this piece of legislation, requires online content-sharing service providers to obtain authorisation from the relevant rights holders prior to making copyright-protected content available on the platform. If they fail to do so, they are liable for the content violating copyright on their platforms unless they make their best effort to: (i) obtain an authorisation, (ii) block unauthorised content and ensure that it remains blocked, and (iii) promptly block or remove unauthorised content once notified [23]. It is argued that article 17 CDSM leads to a situation where nearly any company offering content-sharing services will be required to implement filtering tools in order to avoid the liability regime. Due to the technological limitations of upload-filters explained above, there is a serious risk of over-blocking of a substantial amount of non-infringing content in the EU [9].

Those rules imposed by EU hard-law are complemented by self-regulatory initiatives agreed by the main online platforms, often at the initiative of the European Commission. With regard to online disinformation, which is not always illegal but can be harmful, in 2018, some of the biggest online platforms (Facebook, Google, Twitter, Mozilla and Microsoft) and advertisers, as well as the advertising industry agreed to a Code of Practice on Disinformation. The Code is described as a voluntary, self-regulatory mechanism. The Code's signatories made several commitments. Some of the commitments directly relate to content moderation practices such as: closing false accounts by developing clear policies regarding the identity and misuse of automated bots on their services; investing in technologies to help Internet users make informed decisions when receiving false information (e.g. reliability indicators/trust markers, reporting mechanisms); prioritising relevant and authentic information; or facilitating the finding of alternative content on issues of general interest[24].

In September 2020, the European Commission published its assessment of the Code of practice on disinformation. Numerous positive impacts have been found, including platforms enforcing policies to prevent their services from being used to spread misrepresentative or misleading advertisements; reduced monetization incentives to disseminate disinformation online for economic gain and an introduction of the label for sponsored political ads.

However, a number of shortcomings have also been identified. First, the assessment points out to the fragmented implementation and limited participation (only 16 signatories after almost two years of being in effect), lack of involvement of other relevant stakeholders, in particular from the advertising sector, and a regulatory asymmetry illustrated by the COVID-19 pandemic as two non-signatory platforms - Messenger and WhatsApp - were considered to be serious contributors of the spread of COVID19 disinformation. Second, the absence of relevant key performance indicators (KPIs) to assess the effectiveness of platforms' policies to counter the phenomenon was also pointed out. A lack of commonly shared definitions and more precise commitments combined with lack of enforcement and monitoring mechanisms undermine the Code's impact. The assessment also points out to the lack of adequate complaint procedures and redress mechanisms for wrong content take downs or account suspension following a presumed violation of signatories' disinformation policies and lack of sufficient safeguards to ensure the protection of freedom of expression in practice.

In short, the Code creates a situation which encourages private entities to interfere with the freedom of expression of the Internet users. It creates the incentives to restrict speech that might be critical or controversial but is not illegal under the EU law. Importantly, the following question arises: Who should decide what content is relevant, authentic, accurate and authoritative? And who is responsible if content is mislabeled?

It is important to note that the Code of Practice is currently being revised. The European Commission presented a Guidance to strengthen the Code of Practice on disinformation in May 2021. The Guidance aims to address gaps and shortcomings and create a more transparent, safe and trustworthy online environment. The Guidance also aims at evolving the existing Code of Practice towards a co-regulatory instrument foreseen under the Digital Services Act (DSA).

Moreover, in May 2016, the main online platforms (Facebook, Microsoft, Twitter and YouTube) agreed, at the initiative of the European Commission, an EU Code of Conduct on countering illegal hate speech online and committed to fight the dissemination of illegal hate speech. The Code of

Conduct is another self-regulatory instrument. In particular, the platforms have made a series of commitments to:

- put in place a clear and effective process to review reports/notifications of illegal hate speech and to remove them or make them inaccessible;
- review notifications on the basis of the Community Standards/Guidelines and the national transposition laws, and to review the majority of valid reports within 24 hours;
- encourage the reporting of illegal hate speech by experts, including through partnerships with Civil Society Organisations, so that they can potentially act as trusted reporters; and
- strengthen communication and cooperation between the online platforms and the national authorities, in particular with regard to procedures for submitting notifications.

The Code has however faced certain criticism. First, there is a risk of private censorship practices through the priority application of Community Standards/Guidelines. Second, there is a lack of precision in determining the validity of a notification. There is an absence of appeal mechanisms for users whose content has been withdrawn. The illegal content does not have to be reported to the competent national authorities when removed on the basis of the Community Standards/Guidelines. The 24-hour deadline to review the content makes it impossible for online platforms to meet their commitments and may again lead them to over-blocking (legal) content [25].

Next to the EU regulatory framework, national laws impose additional obligations to moderate some types of illegal content online. In Germany, the Network Enforcement Act (NetzDG) was adopted in June 2017 to improve the enforcement of existing criminal provisions on the Internet and, more specifically, on social networks. In France, two related laws on information manipulation were adopted in December 2018 and a law on online hate speech, the so-called Avia law, was adopted in May 2020. These national rules require more effective moderation, which means more and faster removals of content by online platforms to meet these requirements. Under the threat of high fines, these laws require platforms to limit the dissemination of illegal content as well as harmful content, such as disinformation. The legal compatibility of those national initiatives with the EU legal framework is controversial and was questioned by national constitutional courts.

3.6 Conclusion and future research direction

Efforts to automate content moderation may come at a cost to human rights. Filtering systems, including content moderation systems raise a set of concerns regarding freedom of expression, bias and discrimination issues, due process, as well as surveillance issues that the current legal frameworks have not fully addressed. When combined with social and regulatory pressure on platforms to tackle issues such as disinformation, terrorism content, and hate speech, the application of these tools raises privatized censorship concerns.

There are growing concerns that the measures used by online platforms are not sufficiently effective in moderating illegal content online and in striking an appropriate balance with fundamental human rights. It is said that “the main challenges in moderating illegal content online are linked to the large quantity of online content on platforms, which makes it difficult for users, regulators or moderators to assess all content as well as the fragmentation of laws regarding online content”[24]. The lack of a common definition of ”illegal content” also makes the moderation by platforms more complex as Member States may refer to different definitions[24]. As indicated by Gorwa et al. “as government pressure on major technology companies builds, both companies and legislators seem to hope that technical solutions to difficult content governance puzzles can be found. Under recent regulatory measures like the German NetzDG or the EU Code of Conduct on Hate Speech, platforms are increasingly being bound to a very short time window for content takedowns that effectively necessitates their use of automated systems to detect illegal or otherwise problematic material proactively and at scale”[9].

The Digital Services Act proposed by the European Commission tries to define clear responsibilities and accountability for providers of intermediary services, such as social media and online marketplaces. The DSA proposal is currently in the legislative process and awaits the approval by the European Council and the European Parliament. The main questions which are currently discussed are: how to best create a harmonized framework to tackle illegal content while protecting users' fundamental rights online? What will be the subsequent impact of the proposed obligations on (social) media companies? How will they affect the use of automated tools in content moderation? Task 6.1. will research and provide answers to these questions.

4 Manipulation and synthetic content detection in multimedia (T6.2)

Contributing partners: CERTH, CEA, FhG, QMUL, UPB, UNIFI, UNITN

Detection of manipulated and synthetic content is an emerging research topic of the last decade whose overall objective is the assessment of the authenticity and veracity of multimedia items. Due to the latest advancements in the field of Generative Adversarial Networks (GANs) and Language Models (LMs), we have come to a state where the distinction between real and fake content is becoming increasingly difficult at an alarming rate. Although this may have a beneficial impact on some domains (e.g., art, video games, cinematography), there are several applications that are potentially harmful to individuals, communities, and the society as a whole (e.g., disinformation, identity/financial fraud). This necessitates the development of algorithms and tools that can automatically detect manipulated and synthetic multimedia content to prevent its misuse. The terminology commonly used to describe such content is DeepFake, and the research problem that refers to its detection is called DeepFake detection. Within the AI4Media project, we have developed several DeepFake detection approaches, and we provide tools to WP8 Use Cases (i.e., UC1, UC2, UC3) that facilitate fact-checking and disinformation detection.

Our activities during the first year of the project were focused on three main directions, classified based on the modality used for the detection of fake content. Curated solutions for the different signals are necessary, since the exploitation of the signals' properties is essential in order to achieve high detection performance. In particular, the three main directions and our contributions are:

DeepFake detection and content manipulation based on vision (Section 4.1): we present three methods for detecting manipulated images and videos and one method for image content manipulation. More specifically, UPB proposed a DeepFake detection method for videos using analysis of facial features based on a CNN and LSTM architecture. UNIFI also built a video-based method that exploits the optical flow for the detection of DeepFake videos. CERTH composed a variety of datasets with synthetically generated faces and benchmarked the detection performance of several CNN networks. UNITN proposed an approach for layout-to-image translation based on a novel Double Pooling GAN (DPGAN) with a Double Pooling Module (DPM), which can be exploited to build more robust detection methods.

DeepFake detection based on audio (Section 4.2): two detection methods and a synthetic speech generation method have been developed. FHG-IDMT and CERTH collaborated for the development of an approach for Synthetic speech generation that will be exploited for dataset generation for the training and evaluation of the detection models. Also, CERTH built an Audio DeepFake detector based on deep learning networks, and FHG-IDMT proposed a microphone classification method that contributes to the detection of audio DeepFakes.

DeepFake detection based on text (Section 4.3): we developed an approach for the composition of a dataset with DeepFake tweets and a method to distinguish DeepFake from original tweets. More specifically, CEA trained three deep generative models from Twitter data collected from political and public personalities, they built a dataset in order to train and evaluate a network for DeepFake tweet detection, and investigated the generalisation of the models across accounts.

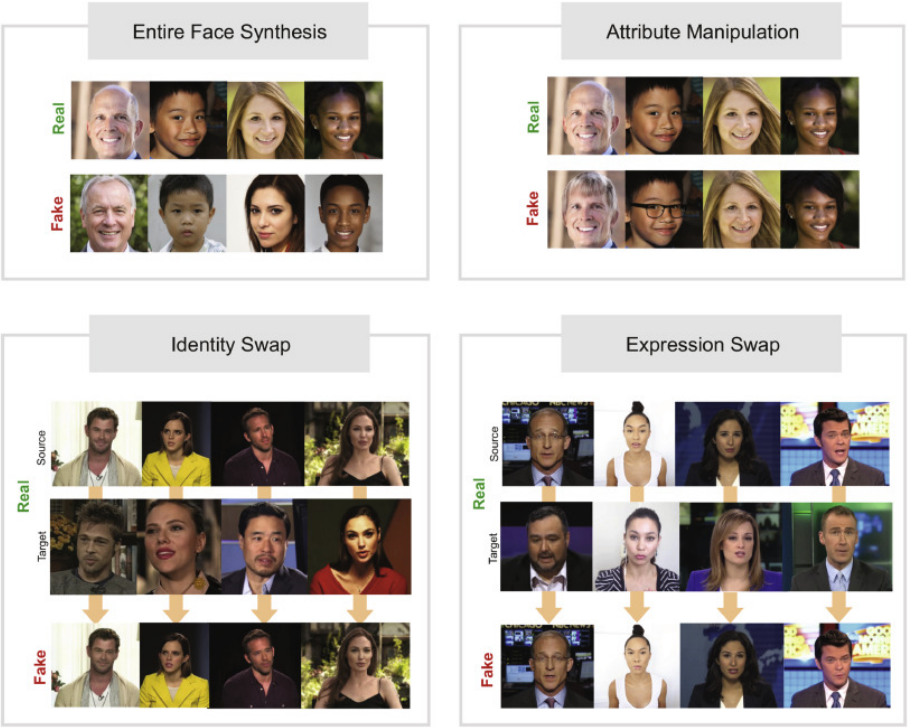


Figure 3. Types of DeepFake manipulation in images and videos. The source of this figure is from [26].

4.1 Vision-based DeepFake detection and generation

This section presents the approaches developed for vision-based DeepFake detection and content manipulation. According to [26], four major manipulation types exist for the generation of DeepFake images and videos (Figure 3): (i) Entire face synthesis, where the images are generated from scratch, (ii) Attribute manipulation, where specific image attributes are altered, (iii) Identity swap, the face of a different person is inserted on top of another, and (iv) Expression swap, where the expression of a person is transferred to another. In Section 4.1.1 and 4.1.2, we present two methods for the detection of identity and expression swap that are based on a CNN-LSTM architecture and an optical flow-based system, respectively. In Section 4.1.3, we train models to detect generated images with entire face synthesis. Finally, in section 4.1.4, we present a novel GAN-based architecture for attribute manipulation.

4.1.1 DeepFake Video Detection with Facial Features and Long-Short Term Memory Deep Networks

DeepFakes have evolved throughout the years, and the content generated by these algorithms seems to be increasingly realistic. Although DeepFake images are just as believable, the most dangerous DeepFakes come in the form of videos. Therefore, there is a pressing need to detect DeepFakes using the temporal dimension. DeepFake detection is a complex task, and that is shown by the diversity and multitude of approaches in the state of the art. At this time, many of the proposed methods work on image level and do not leverage the temporal evolution in the video.

Usually, DeepFakes are formed by generating a whole new facial region. But, there are some cases in which there is no need to falsify the entire face. For example, there are cases when only the mouth region of a person is modified, because it needs to be synchronized with audio. Another example is changing a person's gaze direction, thus only generating their eyes. Because of that, it is crucial for DeepFakes to be detected using only parts of a face.

Our proposed method in [27] tries to solve both of those problems by improving on the work of [28]. The method uses a CNN-LSTM approach to detect DeepFakes from certain facial regions. We use an already proven XceptionNet architecture to extract features from images and a 2-layer LSTM which receives the resulted feature vectors as an input and handles the temporal dimension. For preprocessing, we extract 3 facial regions using facial landmarks: eyes, nose and mouth. We also extract the entire face, but delete the background to eliminate potential error sources.

Figure 4 illustrates the preprocessing pipeline and the model architecture. The model consists of the following: (1) as an input, a sequence of 60 RGB images of 299x299 resolution is passed to the network; (2) the XceptionNet extracts features from every image; (3) the features are passes to a 2-layer LSTM with layer sizes of 256, which outputs a temporal descriptor for the video; (4) a decision layer is used to decide whether the sequence is a DeepFake or not. Two datasets were used for training and evaluation: CelebDF (6,529 videos) [29] and FaceForensics++ (2,000 videos) [30]. We used 80% of the data for training and 20% for evaluation.

Experimental Results For each dataset, we trained 4 different models: one for the full face, and one for each facial region: eyes, nose, mouth. Based on the fact that this is a binary classification problem, the AUC (Area Under Curve) metric was used for evaluation. As a result, we do not need to select a threshold to evaluate performance. In Tables 1 and 2 below, we can see a comparison between the proposed method and similar state of the art methods, on the same datasets.

The experimental results show a significant improvement when using LSTM, as opposed to using frame level information and averaging the results. What is more, we can see that although detecting DeepFakes using only certain facial features yields results that are suboptimal compared to the whole face region, it is possible even for video content. Therefore, we can conclude that detection of DeepFakes where only certain facial regions are attacked is possible. The results below also show that combining the state-of-the-art XceptionNet model with LSTM yields better results than some similar state-of-the-art approaches.

The results for FaceForensics++ [30] are very good, mainly due to the fact that it is a very simple dataset. On the other hand, CelebDF [29] can bring some difficulties. Despite that, our results using a CNN-LSTM architecture show that properly using temporal features can greatly improve performance.

Relevant publications

- `Cristian Stanciu, Bogdan Ionescu, DeepFake Video Detection with Facial Features and Long-Short Term Memory Deep Networks`, ISSCS 2021 [27].
  Zenodo record: `https://zenodo.org/record/5011285#.YZl20dBBwuU`

Relevant software and/or external resources

- The implementation of our work `DeepFake Video Detection with Facial Features and Long-Short Term Memory Deep Networks` can be found in `https://github.com/StanciuC12/deepfake-detection-cnnlstm`.

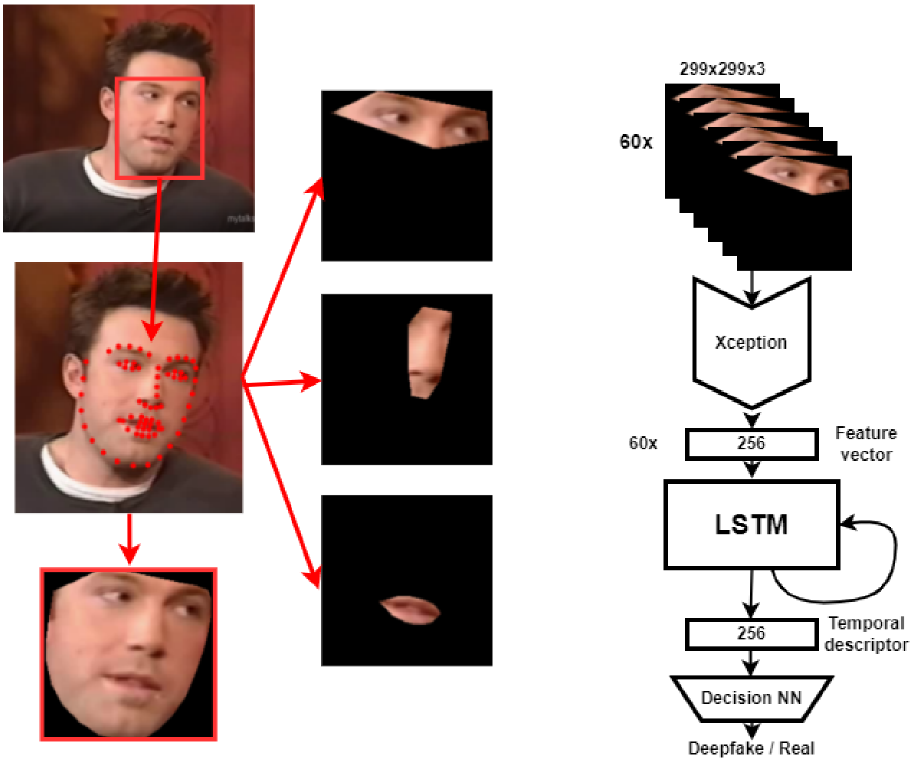


Figure 4. Full deep learning pipeline: Preprocessing steps (Left): (1) Face detection, (2) Extraction of facial landmarks, (3) Extraction and alignment of face and (4) facial regions of interest; Temporal network training (Right): (1) 60 RGB images as input, (2) feature extraction using fine-tuned XceptionNet, (3) 60 sets of features go into the 2 layer LSTM, (4) decision layer, which uses the resulted feature vector

Relevant WP8 Use Cases

This activity relates contributes to the user stories 1A2 (Synthetic Image Detection/Verification), 1A4 (Synthetic Video Detection/Verification), 2A2 (Factchecking Toolbox), and 3C1 (Just-in-Time Content Verification).

Study	Method	CelebDF *AUC[%]*	FF++ *AUC[%]*
Nguyen *et al.* *(2019) [31]*	Capsule Networks	57.50	96.60
Sabir *et al.* *(2019) [32]*	CNN + RNN	-	96.9
Dang *et al.* *(2019) [33]*	CNN + Attention Map	71.2	-
Tolosana *et al.* *(2020) [34]*	Facial Regions Features CNN	83.6	99.4
Proposed [27]	CNN + LSTM	**97.06**	**99.95**

Table 1. AUC comparison between state of the art approaches and our approach applied on the full face.

Face Region	Model	CelebDF *AUC[%]*	FF++ *AUC[%]*
Full Face	Xception-LSTM	97.06	99.95
	Xception [34]	83.60	99.40
Mouth	Xception-LSTM	84.29	98.15
	Xception [34]	65.10	93.90
Nose	Xception-LSTM	75.60	95.35
	Xception [34]	64.90	86.30
Eyes	Xception-LSTM	85.81	98.64
	Xception [34]	77.30	92.70
Late Fusion	Xception-LSTM	86.09	98.46

Table 2. Analysis of the influence of different face regions and improvements of using XceptionNet-LSTM over XceptionNet

4.1.2 DeepFake video detection by means of Optical Flow based CNN

The proposed method is based on the architecture sketched in Figure 5. In the first phase of the pipeline, video frames are processed to estimate the optical flow fields that are then cropped according to a squared box of 300 × 300 pixels containing the speaker face. Such a bounding-box is computed on each frame by using *dlib* [36] face detector. Cropped optical flow (OF) fields are passed as input to a CNN whose final fully connected layer is represented by one output unit followed by a sigmoid activation used for the binary classification of each frame stating if such a frame is tampered or original. In our experiments, the well-known *ResNet50* [37] has been adopted as reference CNN. Different kinds of networks, such as VGG16 [38], had also been considered indeed in our preliminary study in [39]; all of them have substantially provided similar results.

Going into details, the proposed net has been trained on randomly left-right flipped squared patches of size 224 × 224 pixels randomly chosen on the bigger patch of 300 × 300 containing the face, for data augmentation. Specifically for the training phase, we used Adam optimizer with 10^{-4} learning rate, default momentum values and a batch size of 256.

Using pre-trained networks is a reliable technique if not enough data are available for training. We benefit from such initialization as it helps to heavily mitigate the overfitting phenomenon and it determines even a faster convergence. When used with spatial (RGB) frames, we can employ directly *ResNet50* models trained on a large scale dataset such as *ImageNet* but in our case we cannot feed optical flow frames to the available pre-trained networks as their distribution of values is very different from RGB data. To overcome this issue, we decide to bound the range of optical flow values to be the same of RGB frames. Therefore, we firstly clip OF values between -3 and 3 to eliminate outliers (this range is chosen to minimize the information loss) then we scale and discretize OF values into the range [0, 255] with a simple linear transformation.

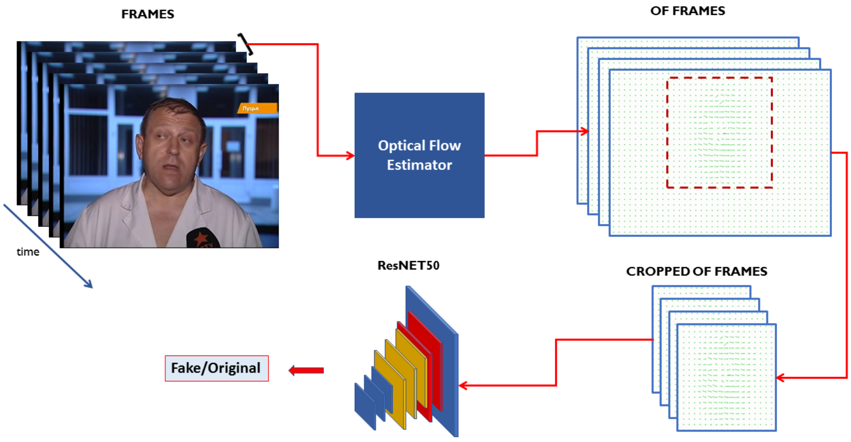


Figure 5. The proposed pipeline. The TV-L1 [35] has been implemented as OF estimator.

As the input dimension of OF frames have only two channels (224 × 224 × 2), this does not match yet the pre-trained network requirements (three color channels), so we proceed to modify the weights of the first convolutional layer of the pre-trained network. Hence, we take the average (along the channels) of the weights of the first convolutional layer and the obtained weights are then replicated to become the new two-channels first layer of the network.

In addition to the proposed optical flow net described above in which the input to the net are OF cropped matrices (as evidenced in Figure 5), we have also investigated if typical training paradigm for DeepFakes detection can benefit from additional motion information provided by the optical flow estimation. For this reason, we have independently trained two different networks having the same architecture, one just with optical flow (OF) frames and the other with spatial (RGB) frames following the idea of most of the state of the art methods. The two contributions derived from OF and from RGB nets are then combined together taken the average of the corresponding classifier outputs (such an approach has been named *MIX* in the experimental results).

Experimental results

Some experimental results are introduced to evaluate the effectiveness of the proposed methodology in different operative contexts. In particular, two distinct scenarios are basically considered: *same-forgery* and *cross-forgery*. The *FaceForensics++ (FF++)* [30] dataset has been used for the experiments; it consists of 1,000 original video sequences that have been manipulated with four face manipulation methods: two graphics rendering approaches *Face2Face* (F2F) and *FaceSwap* (FS) an the other two *DeepFakes* (DF) and *NeuralTextures* (NT), resorting to deep learning methods. DeepFakes and FaceSwap are two different methods for face replacement, while Face2Face and Neural Textures are two facial reenactment systems able to transfer the expressions of a source video to a target video while maintaining the identity of the target person. An amount of 740 videos is used for training, 120 for validation and another 120 for testing. The dataset is composed by three level-of-quality: uncompressed (C0) and compressed using the H.264 codec with a high visual quality (C23) level and a low visual quality (C40). It is worthy saying that the *FF++* dataset has been chosen because it has permitted to properly evaluate the actual performances of the optical flow in a cross-forgery scenario.

OF-based approach in a same-forgery scenario

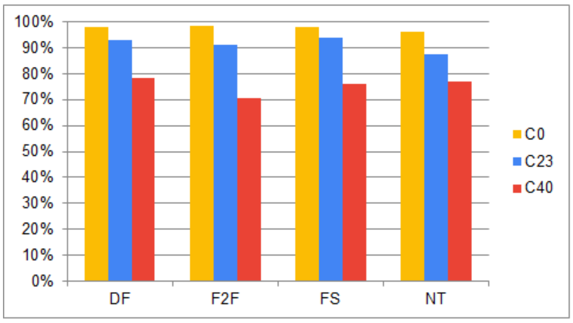


Figure 6. Accuracy (%) of the proposed OF-based approach with respect to the four kinds of forgeries for the three diverse types of video quality: C0 (yellow), C23 (blue) and C40 (red).

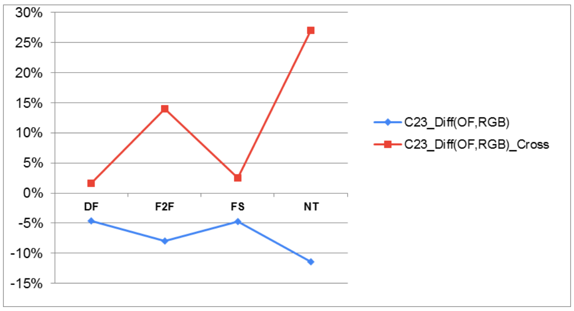


Figure 7. Increment of accuracy (%) in the cross-forgery *(red) and* same-forgery *scenario (blue), respectively (C23 case).*

First of all, we have tested the proposed approach based on optical flow to understand if this new feature is able to highlight a distinctiveness between original and fake videos. In Figure 6, performances, in terms of accuracy, are pictured for the three kinds of video quality available in *FaceForensics++* dataset: C0 (yellow bars), C23 (blue bars) and C40 (red bars). In this case, the network has been trained and tested on the same type of face manipulation (*same-forgery scenario*); it can be appreciated that achieved results are quite satisfactory for all the four forgeries: 97% for C0 and 91% for C23 is averagely obtained respectively, though accuracy is inferior for the circumstance of low video quality (C40) as expected (76% averagely). This shows that optical flow fields are a consistent feature to be learnt in order to detect DeepFakes-like videos.

OF-based approach in a cross-forgery scenario

Going ahead, we have tried to understand if using OF-based features could help in a *cross-forgery* scenario, that is, when a model trained on a certain manipulation is asked to evaluate a video created by resorting to a diverse kind of forgery (e.g. F2F vs. DF, F2F vs. NT and so on) that has never seen before, as it often happens in the real world. This issue is well-known as very challenging and methods such as the one based on RGB frames, usually presenting significant accuracy, drop their performances in this case. In Figures 7 and 8, a comparison, in terms of the achieved accuracy, is reported for the C23 and C40 cases, respectively. In particular, in Figure 7, the red line represents the average accuracy increment obtained by the OF-based method with respect to the one based on RGB spatial frames in the cross-forgery case (the manipulation used to train the model is reported on the x-axis and then it is tested on all the other manipulations). It can be pointed out that such an increment is always positive, sometimes small (as in DF and FS cases) but sometimes higher as in the NT and F2F cases. On the contrary, a decrease is registered for the same-forgery cases (blue line). A similar behavior can be appreciated for the C40 case (see Figure 8) where the increment is above 5% for three out of four forgeries, though with a reduced global effectiveness as expected.

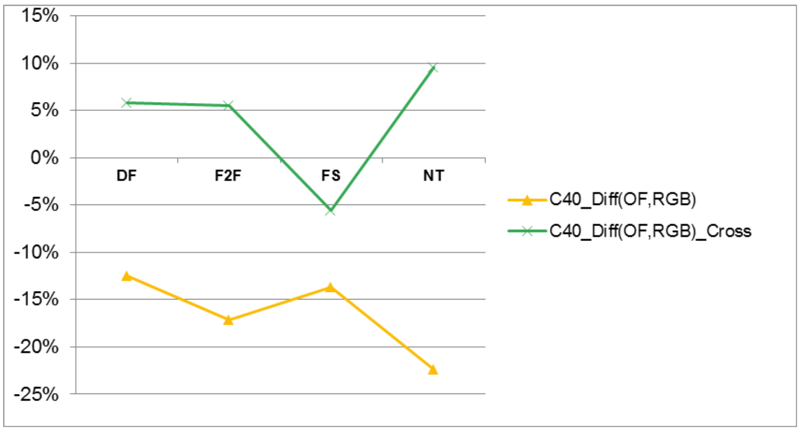


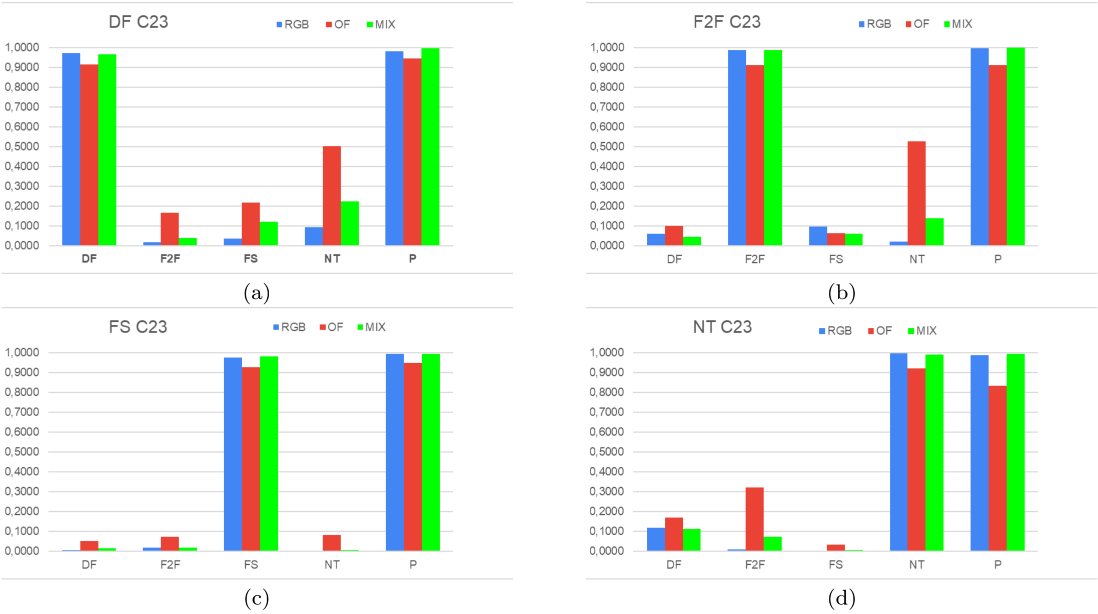
Figure 8. Increment of accuracy (%) in the cross-forgery *(green) and* same-forgery *scenario (yellow), respectively (C40 case).*

However the performance increment in the cross-forgery cases (red and green lines) represents an interesting outcome that encouraged us to combine the two methods, RGB-based and the proposed OF-based named as *MIX* . The two techniques are equally balanced (i.e. with a weight of 0.5 each one) by weighing their outputs simply at the level of the final sigmoid function. The results obtained are discussed in the following.

Results training on one manipulation and testing on all of them

Here, we have investigated more in depth if the proposed idea to resort to OF fields can provide an advantage in a *cross-forgery* scenario. To do this, we have taken into account the following type of experiment: the classifier is binary and has been trained only on one kind of manipulation, for instance *FaceSwap - FS*, and on pristine examples of course, while, during the test phase, it will face, as in a real-world scenario, pristine videos and fake ones, but now these last ones have been generated both through the learnt method and also through other unknown techniques.

In Figure 9, the values of accuracy obtained in the case of C23 dataset are pictured. The four graphs, going from Figure 9 (a) to (d), refer to the cases where the classifier has only been trained on *DF*, *F2F*, *FS* or *NT* manipulation respectively, as indicated by the title of each graph. Every colored bar represents the accuracy achieved by resorting to the frame-based method (RGB, blue bars) and at the OF-based one (OF, red bars) with respect to the diverse kinds of manipulations given at test time (on the x-axis, there are the four DeepFake techniques, while *P* stands for pristine). The green bars values, labelled with MIX, are obtained by means of the combined approach. First of all, by looking at the blue and red bars, it can be observed that the frame-based method (blue) always outperforms the OF-based one (red) when the images to be classified are pristine or fake but generated by the same manipulation learnt during training (i.e. in a *same forgery* scenario). On the contrary, if we check the case when images, crafted with a not-learnt DeepFake technique, are to be evaluated (i.e. in a *cross forgery* scenario), it can be appreciated that the situation is completely inverted: blue bars are always lower than red ones (except for a single case in Figure 9(b) when the model trained on *F2F* is tested on *FS*). This seems to highlight that the OF-based method provide a superior robustness towards fake images created



(a) (b) (c) (d)

Figure 9. ResNet50: cross-forgery experiments on C23 dataset with neural networks trained on DF (a), F2F (b), FS (c) and NT (d). Accuracy achieved is pictured for frame-based method (RGB, blue bars), optical flow-based method (OF, red bars) and the mixed method, (MIX, green bars). P stands for 'pristine'.

with methodologies unknown at training time though it appears slightly under-performing with respect to the frame-based approach in the classical *same forgery* scenario.

On the basis of such a finding, we have tried to understand if this apparent complementary behavior could be composed in order to get a general improvement. So we have mixed, as explained before, the two approaches and if we now look at the green bars in Figure 9, we can effectively observe this phenomenon: the performances in the cases of the *same forgery* scenario (pristine and learnt manipulation) remain as very high as for the frame-based (RGB) method, sometimes even with a slight improvement (e.g. the average accuracy increase of 0.23% for the C23 dataset). What is interesting is the increment in the cases of the *cross forgery* scenario where the accuracy is constantly augmented with respect to the values achieved by the frame-based method alone represented by the blue bars (e.g. the average accuracy increases of 3.14% for the C23 dataset). As expected, the accuracy of the mixed approach (MIX), in these circumstances, is not as high as the OF-based technique by itself but it is generally intermediate between the two (OF and RGB).

Relevant publications

- `Roberto Caldelli, Leonardo Galteri, Irene Amerini and Alberto Del Bimbo, Optical Flow based CNN for detection of unlearnt DeepFake manipulations, Pattern Recognition Letters, 146, pp.31-37` 2021 [40].
  Zenodo record: `https://zenodo.org/record/4707894#.YaTcC9DMJPZ`

Relevant WP8 Use Cases

This activity relates contributes to the user stories 1A2 (Synthetic Image Detection/Verification), 1A4 (Synthetic Video Detection/Verification), 2A2 (Factchecking Toolbox), and 3C1 (Just-in-Time Content Verification).

4.1.3 Detection of synthetically generated images

In this section, we address the problem of the detection of synthetically generated images by GAN-based methods. This is an emerging topic that has drawn the attention of several researchers in the field of multimedia [41]–[44]. Typical approaches employ deep learning algorithms in order to train a deep network, commonly a Convolutional Neural Network (CNN), that ultimately tackles the problem. More precisely, the main advantage of CNN models is that they extract distinctive features for each image that capture relevant information that leads the network to distinguish the real from the GAN-generated images. However, the proposed techniques cannot generalize well to unseen data and fail to build robust detectors that can be effectively applied in datasets that consist of images generated by different Generative Adversarial Network (GAN) architectures.

To this end, in our research, we delve into the performance analysis of such networks evaluated on datasets with various combinations. Our method is based on the pipeline proposed in [41] to train a detection network. Additionally, we compose several datasets with images generated based on two GAN architectures, i.e., StyleGAN2 [45], and ProGAN [46]. We selected generated images based on their GIQA [47] score that measures their quality. We have also downloaded images from the website "This Person Does not Exist" (TPDE)[5], where the generation method is based on StyleGAN2, but the implementation is unknown. Furthermore, the generated images are combined with real ones from the CelebA [48], FFHQ [49], and Human Faces (HF) [50] to constitute an evaluation dataset. To further benchmark the robustness of the models, we draw samples from each dataset to compose a combined one which is considered the more realistic and challenging case. Finally, we experiment with five CNN networks from the state-of-the-art trained and evaluated on our datasets.

Method overview

We base our implementation on the pipeline proposed in [41], where a ResNet50 [37] model is trained with an image set generated with ProGAN combined with various augmentations, and it is then evaluated on different datasets. Its scope was to demonstrate the efficacy of the application of multiple augmentations (e.g., Gaussian Blur or JPEG compression) during training on the given data, which boosts the models' robustness and facilitates the creation of more general detector models. Hence, following this work, we employ augmentations for blurring, sharpening, image corruption, and the color of each image to train our networks. Specifically, during training, a pipeline of transformations is constructed, and random augmentations are applied to the images of the training dataset. The input images are first resized into 256 × 256, and then center-cropped to 224 × 224. Finally, one of the following augmentations is applied:

1. Blurring: Application of two algorithms that emulate blur effect, based on either Motion or Gaussian Blur.
2. Color: Randomly change the values of brightness, contrast, and saturation of images, or transform them into gray-scale.
3. Texture: Manipulation of image texture using Sharpening.
4. Compression: Degradation of image quality based on JPEG compression.
5. Rotation: Rotate input images into different angles.

[5] https://thispersondoesnotexist.com/

Name	Real Images	Fake Images	Size	Scope
T_1	CelebA	StyleGAN2	16K	training
T_2	CelebA	ProGAN	16K	training
D_1	CelebA	StyleGAN2	4K	evaluation
D_2	CelebA	ProGAN	4K	evaluation
D_3	HF	StyleGAN2	4K	evaluation
D_4	HF	ProGAN	4K	evaluation
D_3	HF	TPDE	4K	evaluation
D_6	CelebA + HF + FFHQ	ProGAN + StyleGAN2 + TPDE	12K	evaluation

Table 3. Illustration of datasets' composition. T_i datasets are used for training, and D_i datasets are used for evaluation. Fake images are generated with StyleGAN2 and ProGAN or collected from TPDE, while real images are derived from FFHQ, CelebA, or HF.

6. Arithmetic: Application of noise algorithms that simulates random snow and Add camera sensor noise.

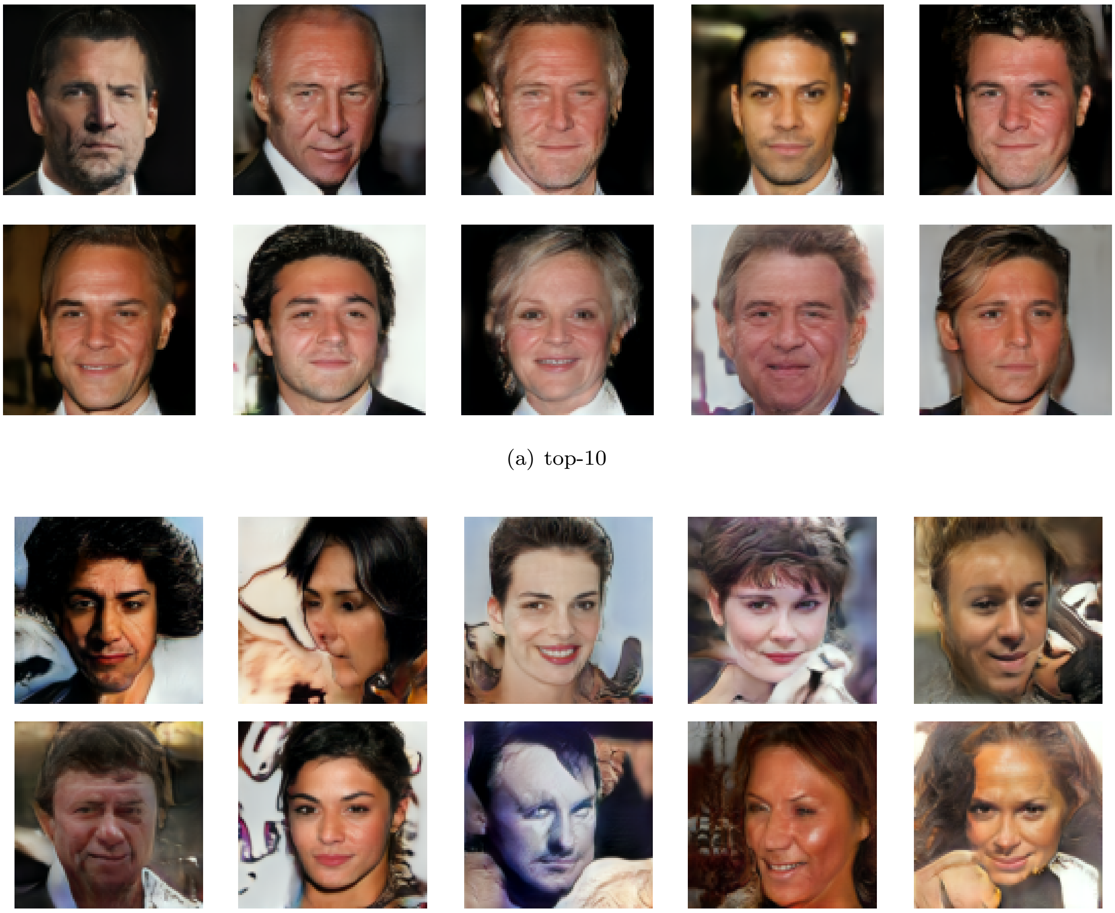
Additionally, we compose several dataset variants in order to evaluate the performance of the trained detector models under different configurations and settings. Specifically, we employ two GAN to generate fake face images, i.e., StyleGAN2 and ProGAN pretrained on the FFHQ and CelebA dataset, respectively. We selected these networks since they are state-of-the-art architectures for image generation. To measure the quality and the photorealism of the generated images, we use the Generated Image Quality Assessment (GIQA) [47] algorithm. There are three main variations of the GIQA algorithm. In this work, the GMM-GIQA is selected, which proceeds as follows: given a dataset of real human face images (i.e., the FFHQ dataset), a Gaussian mixture model is used in order to describe their distribution. More formally, for a given image I the probability is given by:

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} \mathbf{w}^i g(\mathbf{x}|\boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i) \tag{1}$$

where $\boldsymbol{x} = f(I)$ and $f(\cdot)$ denotes a feature extractor function for each image, $\mathbf{w}^i$ are the weights of the mixture model that satisfy the constraint $\sum_{i=1}^{M} \mathbf{w}^i = 1$. The function $g(.)$ are the Gaussian density components of the model parametrized by the mean vector $\boldsymbol{\mu}^i$ and the covariance matrix $\boldsymbol{\Sigma}^i$. In a more general form, the parameters of this model can be notated as $\lambda = \{\boldsymbol{w}^i, \boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i\}$. The estimation of λ is given by the Expectation–Maximization (EM) algorithm [51] (a more general estimation of the maximum likelihood). Then, for each generated image I_g, its quality score S_{GMM} is given by:

$$S_{GMM}(I_g) = p(f(I_g)|\lambda) \tag{2}$$

To this end, the generated images acquired from StyleGAN2 and ProGAN are evaluated based on the GAN algorithm and ranked based on their quality score S_{GMM}. Visual examples of the top and bottom-ranked images are displayed in Figure 10. We retain the top 10K images to work with, and we split them for the training and evaluation of the models. The real images that are used for training and evaluation derive from the datasets CelebA [48], FFHQ [49], and Human Faces (HF) [50] (a web-scraped dataset found in Kaggle). Moreover, two thousand images from the website "This person does not exist" are downloaded in order to use in the evaluation process. We collect images in that way, as this is a popular website that has been used in numerous occasions for generating fake user profile images.



(a) top-10

(b) bottom-10

Figure 10. Example of the top and bottom ten images generated with StyleGAN2 ranked based on their GIQA score.

Table 3 illustrates the composition of the developed datasets. We build two training sets, i.e., the T_1 and T_2 datasets consisting of CelebA real images and StyleGAN2 or ProGAN fake images, respectively. We sample 8K images for each of the two classes, resulting in training datasets with 16K images. Similarly, we construct D_1 and D_2 datasets with the corresponding composition, consisting of 2K images for each class and used for evaluation. The datasets D_3, D_4, D_5, and D_6 were created in order to test the generalization ability of the models and their robustness to different generation algorithms. The first two datasets test the models' performance with real faces from HF, i.e., a different source from the one used for training. The third one serves as a benchmark for the trained models simulating settings where the implementation of the generation algorithm is not available. Finally, the D_6 simulates the most challenging case, combining all fake images from the previous datasets and real images from CelebA, FFHQ, and HF datasets.

We employ five popular network architectures in the field of DeepFake Detection to build the detector models, i.e., Meso4 [52], MesoInception4 [52], Xception [53], ResNet50 [37], and EfficientNet-B4 [54]. The last linear layer of those models is replaced with a linear layer that projects the data

Architecture	D_1	D_2	D_3	D_4	D_5	D_6
Meso4 [52]	0.7513	0.6743	0.7368	0.6585	0.8293	0.6781
MesoInception4 [52]	0.9933	0.9925	0.9548	0.8700	0.8710	0.7878
Xception [53]	0.9925	0.9704	0.9549	0.9322	0.9574	0.8097
ResNet50 [37]	0.9897	0.9894	0.9522	0.9517	0.9559	0.8144
EfficientNet-B4 [54]	0.9942	0.9937	0.8902	0.8884	0.8909	0.7947

Table 4. Accuracy of five different network architectures trained on T_2 dataset on six evaluation datasets.

Architecture	D_1	D_2	D_3	D_4	D_5	D_6
Meso4 [52]	0.8011	0.8028	0.7721	0.7735	0.7775	0.6944
MesoInception4 [52]	0.9890	0.9965	0.8714	0.9618	0.9623	0.8183
Xception [53]	0.9912	0.9994	0.9399	0.9825	0.9430	0.8122
ResNet50 [37]	0.9769	0.9882	0.9462	0.9582	0.9629	0.8165
EfficientNet-B4 [54]	0.9900	0.9957	0.9327	0.9382	0.9377	0.8108

Table 5. Accuracy of five different network architectures trained on T_1 dataset on six evaluation datasets.

to the 1-D space, indicating whether the input images are real or fake. We train our models using the Binary Cross-Entropy loss function. The models are trained for 30 epochs with a batch size of 32 images per batch. Moreover, we employ Adam optimizer with 10^{-3} learning rate, and we apply L_2 regularization with $\lambda = 10^{-5}$ weight decay. Finally, no augmentations are applied during the evaluation process, except the initial resize and center cropping.

Experimental results

Table 4 displays the accuracy of the models trained with the T_1 dataset, which contains real images from the CelebA dataset and fake StyleGAN2 generated. In general, most models achieve high accuracy scores on the evaluation dataset containing images from the same source as the training set, but their performance drops when applied to datasets from different sources. More precisely, it is evident that the accuracy drops slightly by changing the set of fake images but maintaining the same set of real images, which derives from the comparison of the models' scores on D_1 and D_2. There is a significant impact on the performance when real images from a different source are used, i.e., the accuracy of each model drops more than 4% on D_3, D_4, and D_5; however, it remains almost the same irrespective of the set of fake images. In D_6, which is the most challenging case, the performance of all models decreases by almost 20%. In terms of the network architectures, ResNet50 is the most robust being among the top performing architectures, achieving the best accuracy on D_6 with 81.44% demonstrating good generalization ability. On the other hand, Meso4 has the worst performance out of the five networks with more than 10% absolute difference in accuracy.

Furthermore, Table 5 presents the accuracy of the five networks trained with the dataset T_2 containing real images from the CelebA dataset and fake images generated based on ProGAN. Similar conclusions derive from this experiment. Specifically, almost all models demonstrate very high performance when evaluated on the datasets with images from the CelebA, as in D_1 and D_2. However, their accuracy drops when real images derive from HF, as in D_3, D_4, and D_5. In the case of D_6, all models' performance significantly drops, with the MesoInception4 reporting the best score with 81.83%. Also, Meso4 reports the worst performance among the benchmarked networks with these settings as well. Comparing the results in Table 4 and Table 5, we may conclude that the use of different GAN models for the generation of the fake images does not have considerable impact on the performance of the networks.

Relevant software and/or external resources

- For this research, we work with the GIQA implementation provided in https://github.com/cientgu/GIQA.
- Also, we used StyleGAN2 and ProGAN implementations provided in https://github.com/genforce/interfacegan.

Relevant WP8 Use Cases

CERTH provides two tools to the WP8 Use Cases, i.e., Image/Video DeepFake Detector for detection of DeepFake faces in multimedia, and Image Verification Assistant for tampering localization. In particular, the tools contribute to the user stories 1A2 (Synthetic Image Detection/Verification), 1A4 (Synthetic Video Detection/Verification), 2A2 (Factchecking Toolbox), and 3C1 (Just-in-Time Content Verification).

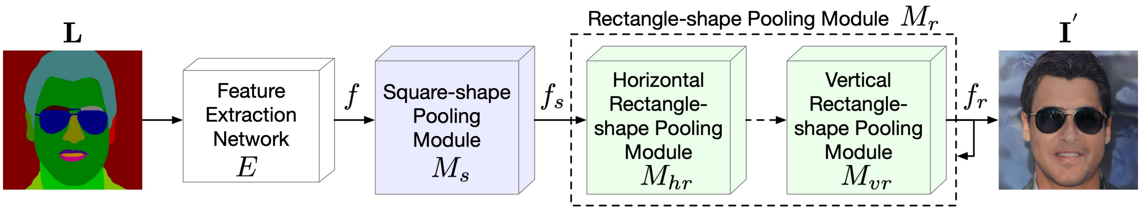

L
Feature Extraction Network E
f
Square-shape Pooling Module M_s
f_s
Rectangle-shape Pooling Module M_r
Horizontal Rectangle-shape Pooling Module M_{hr}
Vertical Rectangle-shape Pooling Module M_{vr}
f_r
$\mathbf{I}'$

Figure 11. Overview of the generator G of our proposed DPGAN, which consists of a feature extraction network E, a Square-shape Pooling Module M_s, and a Rectangle-shape Pooling Module M_r. All components are trained in an end-to-end fashion so that M_s and M_r can benefit from each other by capturing both long-range and short-range semantic dependencies.

4.1.4 Layout-to-Image Translation with Double Pooling Generative Adversarial Networks

The goal of our research is addressing the challenging layout-to-image translation task, which has a wide range of real-world applications such as content generation and image editing [55]–[57]. This task has been widely investigated in recent years [57]–[62]. For example, Park et al. [58] proposed the GauGAN model with a novel spatially-adaptive normalization to generate realistic images from semantic layouts. Tang et al. [62] proposed the LGGAN framework with a novel local generator for generating realistic small objects and detailed local texture. Despite the interesting exploration of these methods, we can still observe blurriness and artifacts in their generated results because the existing methods lack an effective semantic dependency modeling to maintain the semantic information of the input layout, causing intra-object semantic inconsistencies. To solve this limitation, we propose a novel Double Pooling GAN (DPGAN) and a novel Double Pooling Module (DPM).

The proposed Dual Pooling GANs (DPGAN) consists of a generator G and discriminator D. An illustration of the proposed generator G is shown in Figure 11, which mainly consists of three components, i.e., a feature extraction network E extracting deep features from the input layout $\mathbf{L}$, a Square-shape Pooling Module (SPM) modeling short-range and local semantic dependencies, and a Rectangle-shape Pooling Module (RPM) capturing long-range and global semantic dependencies from both horizontal and vertical directions. SPM and RPM together form our proposed Double Pooling Module (DPM). Moreover, we propose seven image-level and feature-level fusion methods to combine both the outputs of SPM and RPM.

Feature Extraction Network. As shown in Figure 11, the network E receives the semantic layout $\mathbf{L}$ as input and outputs the deep feature f, which can be formulated as,

$$f = E(\mathbf{L}). \tag{3}$$

Then, f is fed into the proposed SPM and RPM for learning short-range and long-rang semantic dependencies, respectively.

Square-Shape Pooling Module. Existing layout-to-image translation methods such as [58], [59], [62], [63] directly use deep features generated by convolutional operations, leading to limited effective fields-of-views and thus generating different textures in the pixels with the same label. To model short-range and local semantic dependencies over the deep feature f, we propose a Square-shape Pooling Module (SPM). Note that the idea of the proposed SPM is inspired by the pyramid pooling module proposed in [64] and we extend the original module used in image segmentation to a completely different image generation task. The framework of SPM is elaborated in Figure 12.

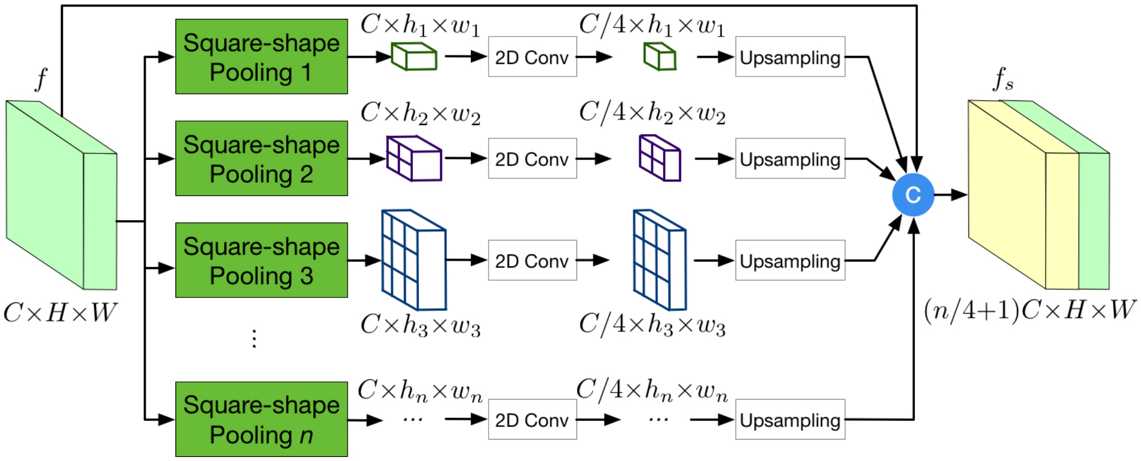

f
$C{\times}H{\times}W$
Square-shape Pooling 1
Square-shape Pooling 2
Square-shape Pooling 3
⋮
Square-shape Pooling n
$C{\times}h_1{\times}w_1$
$C{\times}h_2{\times}w_2$
$C{\times}h_3{\times}w_3$
$C{\times}h_n{\times}w_n$
2D Conv
2D Conv
2D Conv
2D Conv
$C/4{\times}h_1{\times}w_1$
$C/4{\times}h_2{\times}w_2$
$C/4{\times}h_3{\times}w_3$
$C/4{\times}h_n{\times}w_n$
Upsampling
Upsampling
Upsampling
Upsampling
C
f_s
$(n/4{+}1)C{\times}H{\times}W$

Figure 12. The proposed Square-shape Pooling Module (SPM) which aims to capture short-range and local semantic dependencies. Our SPM is a n-level pooling module with different square-kernel size, i.e., (h_1, w_1), (h_2, w_2), $\cdots$, (h_n, w_n), where $\{h_i{=}w_i\}_{i=1}^{n}$. The symbol ⓒ denotes channel-wise concatenation.

Rectangle-Shape Pooling Module. The proposed SPM captures only short-range semantic dependencies. To capture long-range and global semantic dependencies, we can increase the kernel size of the square pooling. However, this inevitably incorporates lots of irrelevant regions when processing rectangle-shaped and narrow objects.

To alleviate this limitation, we propose a novel Rectangle-shape Pooling Module (RPM) (see Figure 13), which aims to capture long-range and global semantic dependencies from both horizontal and vertical directions. The idea of the proposed RPM is inspired by the strip pooling module proposed in [65] and the framework of RPM is illustrated in Figure 13. It consists of a Horizontal Rectangle-shape Pooling Module (HRPM) and a Vertical Rectangle-shape Pooling Module (VRPM). HRPM captures long-range dependencies from horizontal and narrow objects, while VRPM captures long-range correlations from vertical and narrow objects.

Experimental results

Datasets. We follow GauGAN [58] and firstly conduct experiments on Cityscapes [66] and ADE20K [67] datasets. Cityscapes contains street scene images, and ADE20K contains both indoor and outdoor scenes. To further evaluate the robustness of our method, we conduct experiments on three more datasets with diverse scenarios, i.e., DeepFashion [68], CelebAMask-HQ [69], and Facades [70]. DeepFashion contains human body images, CelebAMask-HQ contains human facial images, and Facades contains facade images with diverse architectural styles. Experiments are conducted using different image resolutions to validate that our DPGAN can also generate high-resolution images, i.e., ADE20K (256×256), DeepFashion (256×256), Cityscapes (512×256), Facades (512×512), and CelebAMask-HQ (512×512).

Evaluation Metrics. We follow GauGAN [58] and adopt mean Intersection-over-Union (mIoU), pixel accuracy (Acc), and Fréchet Inception Distance (FID) [71] as the evaluation metrics on Cityscapes and ADE20K. For DeepFashion, CelebAMask-HQ, and Facades datasets, we use FID and Learned Perceptual Image Patch Similarity (LPIPS) [72] to evaluate the quality of the generated images.

We adopt GauGAN [58] as our backbone and insert the proposed Double Pooling Module (DPM) before the last convolution layer to form our final model, i.e., DPGAN.

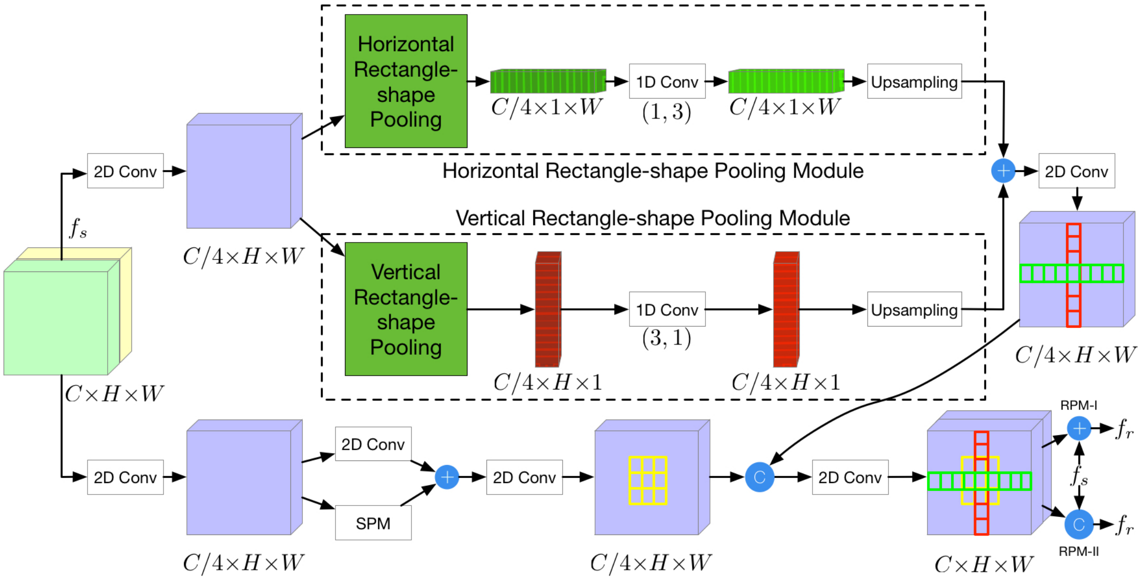


Figure 13. The proposed Rectangle-shape Pooling Module (RPM) which consists of a Horizontal Rectangle-shape Pooling Module (HRPM) and a Vertical Rectangle-shape Pooling Module (VRPM), aiming to capture long-range and global semantic dependencies from horizontal and vertical direction, respectively. The yellow, green, and red grids represent short-dependency, horizontal long-dependency, and vertical long-dependency, respectively. The symbols ⊕, and Ⓒ denote element-wise addition, and channel-wise concatenation, respectively.

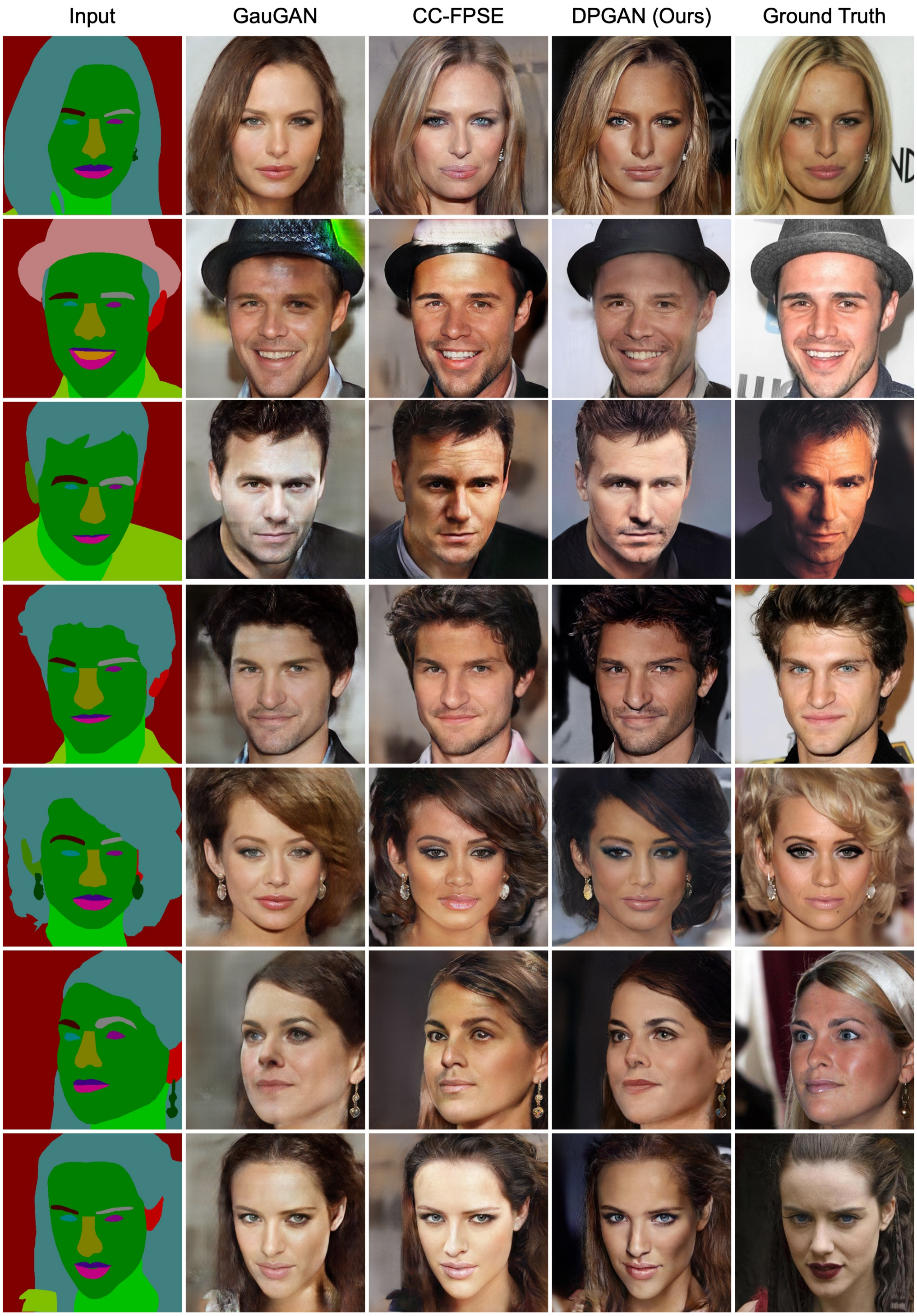


Figure 14. Qualitative comparison on CelebAMask-HQ. From left to right: Input, GauGAN [58], CC-FPSE [59], DPGAN (Ours), and Ground Truth. We see than DPGAN generates more convincing details than GauGAN and CC-FPSE, e.g., the hair, the hat, and the face skin in the first, second, and third row, respectively.

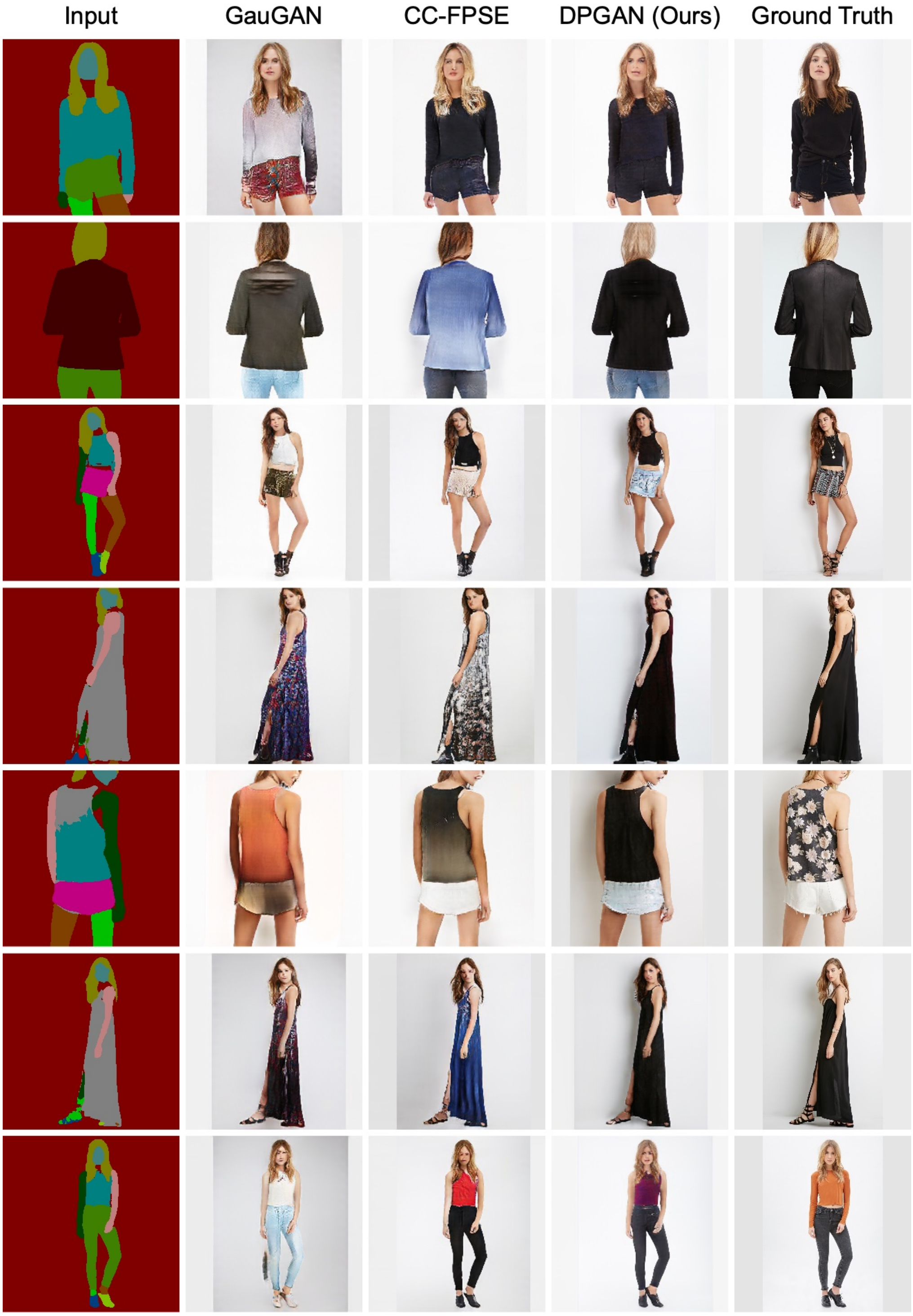


Figure 15. Qualitative comparison on DeepFashion. From left to right: Input, GauGAN [58], CC-FPSE [59], DPGAN (Ours), and Ground Truth. We see that DPGAN generates more photo-realistic clothes than GauGAN and CC-FPSE.

Table 6. User study. The numbers indicate the percentage of users who favor the results of our proposed DPGAN over the competing methods.

AMT ↑	Cityscapes	ADE20K	DeepFashion	Facades	CelebAMask-HQ
Ours vs. GauGAN [58]	65.78	68.72	66.85	67.54	69.91
Ours vs. CC-FPSE [59]	62.21	64.36	63.16	64.54	67.18

Table 7. Quantitative comparison of different methods on DeepFashion, Facades, and CelebAMask-HQ.

Method	DeepFashion		Facades		CelebAMask-HQ	
	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓	FID ↓	LPIPS ↓
GauGAN	22.8	0.2476	116.8	**0.5437**	42.2	0.4870
+ DPM (Ours)	**20.8**	**0.2455**	**115.1**	0.5503	**25.1**	**0.4823**

Qualitative Comparisons. We first compare the proposed DPGAN with GauGAN [58] and CC-FPSE [59] on DeepFashion, CelebAMask-HQ, and Facades datasets. Note that we used the source code provided by the authors to generate the results of GauGAN and CC-FPSE on these three datasets for fair comparisons. Visualization results are shown in Figures 14 and 15. We can see that the proposed DPGAN generates more photo-realistic and semantically-consistent results than both GauGAN and CC-FPSE.

User Study. We follow the same evaluation protocol of GauGAN and also perform a user study. The results compared with GauGAN and CC-FPSE are shown in Table 6. We see that users strongly favor the results generated by our proposed DPGAN on all datasets, further validating that the generated images by our DPGAN are more photo-realistic.

Quantitative Comparisons. Although the user study is more suitable for evaluating the quality of the generated image in this task, we also follow GauGAN and use mIoU, Acc, FID, and LPIPS for quantitative evaluation. The results compared with several leading methods are shown in Table 7. Firstly, we observe that the proposed DPGAN achieves the best results compared with GauGAN on DeepFashion, CelebAMask-HQ, and Facades datasets, as shown in Table 7.

Generalization of DPM. The proposed Double Pooling Module (DPM) is general and can be seamlessly integrated into any existing GAN-based architecture to improve the image translation performance.

Relevant publications

- `H. Tang, N. Sebe, Layout-to-Image Translation with Double Pooling Generative Adversarial Networks, IEEE Transactions on Image Processing, 30:7903-7913, September 2021.` [73].
  Zenodo record: `https://zenodo.org/record/5520463`.

Relevant software and/or external resources

- The implementation of our work "Layout-to-Image Translation with Double Pooling Generative Adversarial Networks" can be found in `https://github.com/Ha0Tang/DPGAN`.

Relevant WP8 Use Cases

Our tool for layout to image translation contributes to use cases (a) 3A3 (Archive exploitation) and 4C1 (Still Image analysis) by providing solutions to analyze visual content, and (b) 7A3 (Organisation of video collections) by supporting the organization of image and video collections.

4.2 Audio-based DeepFake detection

This section focuses on current AI4Media activities related to the detection of audio DeepFakes. The methods presented hereafter thus apply to both audio files and audio streams of video files.

In Section 4.2.1 we describe how to generate a coherent set of audio DeepFakes to be used for training and testing of current and future DeepFake detection algorithms. In Section 4.2.2, we present a first approach to distinguish real speech recorded with a device from synthetic speech generated by text-to-speech algorithms. Lastly, in Section 4.2.3 we report a novel method to distinguish which device was used to record an audio file in the presence of non-negligible background noise.

4.2.1 Synthetic speech generation

Partners participating in the audio analysis task agreed that a necessary prerequisite to DeepFake detection is their generation and collection in an appropriate dataset. The generation of synthetic speech is not only necessary to have a better understanding of the technologies related to audio DeepFakes, but also to ensure that the data are in line with the ones expected by the requirements of the use cases connected to T6.2.

Modern algorithms for Text-to-Speech (TTS) applications are mostly based on the combination of a Feature Generation (FG) stage, in which an input text is converted to a spectral representation (e.g., linear or mel spectrogram), and a Vocoding (V) step, in which the intermediate representation is converted into the final waveform – as depicted in Figure 16. Examples of state-of-the-art in this domain are, among others, the FastSpeech2 acoustic feature generator [74], the HiFi-GAN neural vocoder [75], and the MelGAN neural vocoder [76].

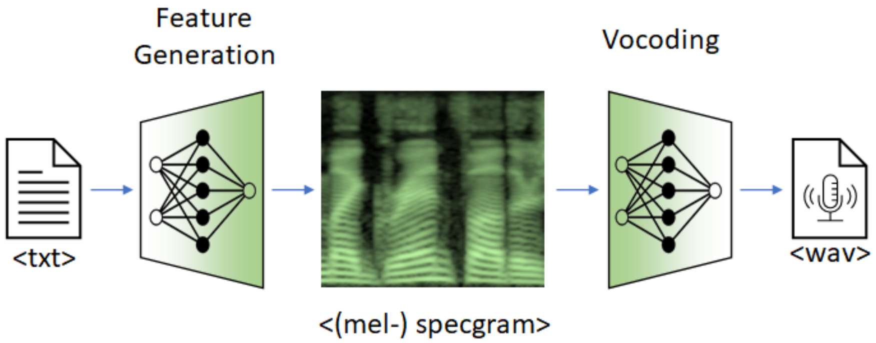


Figure 16. High level schema of neural text-to-speech applications.

Due to the fast pace at which both FG and V networks are being proposed, no comprehensive and high-quality dataset of synthetic audio has been released to the public. Such a dataset would need to (i) include high quality content for both the natural and the synthetic speech examples, (ii) include both short and *long* utterances lasting more than only 3-4 seconds, and (iii) include both synthetic and natural examples for *more* than one person.

At present, only two datasets address synthetic speech detection: the FoR (Fake-or-Real) [77] and the ASVspoof [78] dataset. However, they both lack the requirements mentioned above. CERTH and FHG-IDMT thus decided to collaborate and generate a *common* dataset for synthetic speech to be used by both partners for the training and testing of the corresponding detectors. To create the dataset, CERTH and FHG-IDMT agreed on selecting networks for TTS applications which not only represent the current state-of-the-art of the domain, but are also publicly available on the internet and easily deployable – i.e., they are the ones which are more likely to be used by malicious actors generating and manipulating content.

The networks selected up to now, listed in Table 8, were wrapped in separate Docker images all using common I/O formats, in order to ensure easy composition of the architectures. The resulting code is available online[6] to both partners, and can be enriched with new architectures at will. The dataset resulting from this joint contribution is going to be published on Zenodo, and described by an accompanying peer-reviewed publication.

Table 8. V(ocoding) and F(eature) G(eneration) architectures for the future AI4Media dataset of synthetic speech

Architeture		Type	Online source
HiFi-GAN	[75]	V	https://github.com/jik876/hifi-gan
MelGAN	[76]	V	https://github.com/seungwonpark/melgan
WaveFlow	[79]	V	https://github.com/PaddlePaddle/Parakeet/tree/v0.3.0
WaveGlow	[80]	V	https://github.com/NVIDIA/waveglow
FastSpeech2	[74]	FG	https://github.com/ming024/FastSpeech2
Tacotron2	[81]	FG	https://github.com/NVIDIA/tacotron2
TransformerTTS	[82]	FG	https://github.com/as-ideas/TransformerTTS

This joint activity brought several benefits:

- Set the basis for a further collaboration for what concerns the *detection* of synthetic speech;
- Provided both partners with know-how in the *generation* of synthetic speech;
- Provided a common set for *benchmarking* performances of both generation and detection methods;
- Led to a concrete and agreed definition of *requirements* of both test and training data;
- Is going to lead to a shared publication, increasing the research output and reach of the AI4Media project.

Experimental results

The minimum execution time for inference was calculated after ten repetitions. Testing was conducted on a machine with NVIDIA 1050Ti GPU and Ryzen 1600 CPU and the inference of each file was executed sequentially, without utilizing batch inference capabilities of some of the models used (Table 9). Example Mel spectrograms extracted from inferred audio with the use of a text phrase sample are depicted in Figure 17.

Relevant software and/or external resources

Software related to synthetic speech generation is collected collaboratively in the aforementioned shared GitLab repository, a screenshot of which is depicted in Figure 18.

[6] https://gitlab.cc-asp.fraunhofer.de/groups/ai4media

Table 9. TTS pipelines inference time results

Pipeline Model	Inference Time(seconds)		
	Feature Extractor	Vocoder	Total
FastSpeech2 & WaveFlow	6.175	26.567	32.742
Tacotron2 & WaveFlow	1.751	27.312	29.063
FastSpeech2 & WaveGlow	6.175	27.573	33.748
Tacotron2 & HiFi-GAN	1.751	0.144	1.895
FastSpeech2 & HiFi-GAN	6.175	0.144	6.319
Tacotron2 & MelGAN	1.751	0.853	2.604
FastSpeech2 & MelGAN	6.175	0.853	7.028

Relevant WP8 Use Cases
This activity is a necessary prerequisite for detection of synthetic speech, and thus contributes to the following AI4Media use-case requirements:

- 1A3 – Synthetic Audio Detection/Verification
- 1A4 – Synthetic Video Detection/Verification
- 2A2 – Factchecking Toolbox
- 3C1 – Just-in-Time Content Verification

4.2.2 Audio DeepFake detector

In the proposed methodology, various audio feature extraction methods were tested to address the problem of selecting the optimal audio feature to be used for synthetic speech detection. The performance of each feature was evaluated through the training of two state-of-the-art Deep Neural Network (DNN) models, VGG16 [83] and MLP-Mixer [84] . The Fake or Real (FoR) dataset, an audio dataset with real and synthetic speech utterances, was used for the training process.

Dataset – The FoR dataset[77] is composed of more than 87,000 synthetic utterances and 117,000 real utterances of speech. The synthetic utterances are produced by a variety of commercial and open source state-of-the-art speech synthesis algorithms, utilizing a dataset[7] of English phrases translated to French. The real utterances are a compilation of audio recordings provided by common open source audio datasets, namely Arctic Dataset[8], LJSpeech[9] and VoxForge[10]. Four versions of the dataset are available, each following a different preprocessing methodology to cover a wide variety of machine learning applications. For our experiments, the FoR-2seconds version was chosen, containing a total of 17,870 speech utterances with a sample rate of 16 kHz, that are truncated at the two-second mark and are evenly distributed for class and gender. The division of the dataset in training, validation and testing is as follows: 77.3% of the samples in training, 15.58% in validation and 6.68% in testing. Furthermore, all the subsets maintain gender and class balance while the testing set includes audio samples from an unseen TTS algorithm and unseen real voices. This ensures that the deep learning models developed can be evaluated on their ability to perform well on real life scenarios, where the audio samples have different characteristics than the ones they were trained on.

[7] https://www.kaggle.com/percevalw/englishfrench-translations
[8] http://festvox.org/cmu arctic/
[9] https://keithito.com/LJ-Speech-Dataset/
[10] http://www.voxforge.org/

Models – In recent years, a plethora of different neural network architectures have achieved state-of-the-art results in classification tasks of image data. CNNs show consistent performance in domains where visual data are used, followed by Visual Transformers which build upon the achievements of Transformers in the field of Natural Language Processing. However, the main focus of our experiments is the evaluation of audio features for fake audio detection, hence, an extensive testing of various DNNs is beyond the scope of this study. Two DNNs were evaluated , namely VGG16 [83] and MLP-Mixer [84].

VGG16 is a CNN proposed by K. Simonyan and A. Zisserman of the University of Oxford that won the 2014 ImageNet Large Scale Visual Recognition Challenge. The main idea behind the VGG16 architecture is the use of small 3×3 convolution layers with fixed stride and padding of 1 pixel, followed by a max pooling layer of 2×2 pixel window with a stride of 2 pixels. This order of convolution and max-pooling layers is maintained throughout the architecture. The final classifier, preceded by a flattering layer, includes three Fully Connected Layers and a softmax for output. In our experiments, we adapted the classifier, to fit the binary classification problem at hand, by using two Fully Connected Layers with 1024 neurons and a simple 1 neuron layer with sigmoid activation function for output.

MLP-Mixer is a new model proposed by researchers from Google based on the recent achievements of Transformers in both Natural Language Processing and Computer Vision problems. The MLP-Mixer architecture consists of two MultiLayer Perceptron (MLP) blocks that use Gaussian Error Linear Unit activation function and are connected sequentially, one for token-mixing and one for channel-mixing. Image patches, which are also referred to as tokens, is an idea adopted by Visual Transformers to replace convolution filters used in CNN models. An input image is converted into a sequence S of non-overlapping image patches, each one projected into a hidden dimension C, resulting in a two-dimensional real input table X of size $S \times C$. The first MLP block is the token-mixing MLP which acts on columns of X to mix tokens of the same channel and encode the cross-location information. The second MLP block is the channel-mixing MLP which acts on rows of X, thus ,encoding the per-location information of tokens of different channels. Lastly, a global average pooling layer and a linear classifier outputs the final classification result. The main advantage of MLP-Mixer architecture is its computational complexity, which is linear in the number of image patches, compared to Visual Transformers' complexity that scales quadratically.

Feature Extraction

In the training process of a DNN model based on audio data, the feature extraction methodology plays the most crucial role in the overall performance of the model. The first feature extraction methods in literature focused on the use of raw audio data in the time domain [85]. In this approach, the three-dimensional audio signal, that consists of frequency, amplitude and time, is modelled through temporal-based features which are learned by the model and not handcrafted during the preprocessing phase. The second approach, that we focus on this study, consists of methods that compute time-frequency representations of the signal such as Short-Time Fourier Transform (STFT) spectrograms, Mel-Frequency Cepstral Coefficients (MFCC) and Mel spectrograms[86]. The features are fed in our models as magnitude spectrogram representations in image format or as spectral energies in a two-dimensional matrix.

For the single-channel representation of the audio samples, two magnitude representations were used, STFT and Mel spectrograms. The STFT spectrogram is a visual representation of the normalized, square magnitude (power spectrum) of the STFT coefficients produced via the computation of Fourier Transform for successive frames in an audio signal. Mel-Spectrogram follows the same procedure but the frequency axis is scaled to the Mel scale (an approximation of the nonlinear scaling of the frequencies as perceived by humans). The STFT spectrogram representation resulted in a 251×512 matrix (251 STFT time frames and 512 discrete frequencies up to the Nyquist frequency while the Mel-spectrogram resulted in a 251 × 256 matrix (256 Mel frequency bins). The images for each spectrogram were saved as grayscale images and the spectrum energies as Numpy arrays.

In the case of multi-dimensional spectrogram representation, the grayscale images of STFT, Mel and MFCC spectrograms (with 60 MFCC bands) were reshaped to 256 × 512, keeping the aspect ratio, and then stacked, returning a 256 × 512 × 3 matrix. The three-dimensional matrices were saved as RGB images and were fed into the models. A sample of the single-channel and the multi-dimensional magnitude spectrogram representations can be seen in Figure 19.

Experimental results VGG16 and MLP-Mixer were trained for each of the aforementioned audio features and the results were evaluated on test set provided by the FoR dataset. For the VGG16 model, the classifier was replaced with two Fully Connected layers of 1,024 neurons, with ReLU activation function and a single neuron layer with sigmoid activation function for the binary classification output. Input data were loaded with a batch size of 16 and and the Stochastic Gradient Descent (SGD) optimizer was selected with a learning rate of 0.01. For MLP-Mixer, the parameters selected are as follows: number of blocks=2, patch size=8, hidden dimension=32, token mlp dimension=64, channels mlp dimension=128, optimizer = rmsprop and learning rate = 0.01.

During the experimentation, it was observed that both models were converging in solutions with 95% accuracy on the validation set in the first three to five epochs. As a result, an early stopping was applied to both models to stop further training when the the accuracy surpassed the 95% accuracy mark. The early stopping resulted in improved benchmark scores on the test set, compared to prior experimentation with an early stopping that halted training when there were no improvements higher than 0.01 after five epochs. The latter early stopping would train models for more than 50 epochs, increasing the validation set accuracy marginally but returning very low benchmark scores.

The classification benchmark score for every combination of audio feature and model architecture can be seen in Table 10. VGG16 outperformed MLP-Mixer in all scenarios, with the only advantage of MLP-Mixer architecture being to be limited in very fast training times (6 times fast than VGG16). Different parameterization and further scaling of MLP-Mixer did not yield any improvements. Moreover, MLP-Mixer failed to converge when trained with the audio multi-feature spectrogram, Mel and STFT spectral energies.

From a comparative analysis of the different audio features scores, in Table 10, it can be inferred that the MLP-Mixer's under-performance is due to the architecture of the model, since all features achieve similar results. The best performing features for VGG16 are Mel Spectral energies and STFT spectrograms, achieving high scores in all four metrics. Mel Spectrograms and STFT spectral energies have comparable results with the aforementioned features, albeit achieving lower recall and F1-score. In a real life scenario, the fake speech synthesis detection system would be expected to minimize the number of false negative predictions with the aim of maximizing protection against possible adverse attacks. False positive predictions are of lesser importance, since there can be more than one ways to evaluate audio samples, such as human experts. Therefore, recall is the primary metric of choice for the current analysis.

Relevant WP8 Use Cases

This activity addresses the detection of synthetic speech, and thus contributes to the following use-cases:

- 1A3 – Synthetic Audio Detection/Verification
- 1A4 – Synthetic Video Detection/Verification
- 2A2 – Factchecking Toolbox

Table 10. Performance of VGG16 and MLP-Mixer trained with different audio features.

Model - Feature	Accuracy	Precision	Recall	F1-Score
VGG16 - Mel spectral energies	0.94	0.95	0.94	0.94
VGG16 - Mel Spectrogram	0.88	0.97	0.79	0.87
VGG16 - STFT spectral energies	0.91	1	0.83	0.90
VGG16 - STFT Spectrogram	0.95	1	0.90	0.94
VGG16 - Multi-Features	0.69	0.98	0.38	0.55
MLP-Mixer - Mel Spectrogram	0.68	0.90	0.41	0.57
MLP-Mixer - STFT Spectrogram	0.69	0.98	0.38	0.55

- 3C1 – Just-in-Time Content Verification

4.2.3 Microphone classification in noisy environments

Microphone classification is a classic problem related to the authenticity analysis of audio recordings. Given an unlabeled audio signal $x(t)$ and a set $\mathcal{X} = \{x_i\}$ of *known* recording devices, the goal of microphone classification is to determine which device x_i was used to acquire the audio signal under investigation, given a device fingerprint extracted from the device. In terms of machine learning, the operation consists a closed-set classification task, in which a pre-trained model classifier predicts a label x_i, given a feature vector extracted from the input signal $x(t)$, as shown in Figure 20.

The problem has been thoroughly addressed in the literature, e.g. in [87]–[91], by creating device fingerprints which model the microphone frequency response of the device, denoted by $F_{\text{mic}}(f)$ in eq. (4)

$$x(t) = \int F_{\text{mic}}(f) \cdot [S(f) + N_{\text{env}}(f)]\, df. \tag{4}$$

In the equation above, $S(f)$ denotes the input (speech) signal in the frequency domain, $N_{\text{env}}(f)$ denotes any environmental additive noise in the frequency domain, and $x(t)$ the resulting audio signal under analysis. State-of-the-art methods work remarkably well in nearly *noiseless* conditions – i.e., by assuming $N_{\text{env}}(f) = 0$ in eq. (4) – but suffer from a great performance decrease whenever applied to *noisy* conditions, when the assumption does not hold; e.g., the accuracy of the algorithm we selected as baseline [88], [89] drops dramatically from about 96% to about 22%.

In our method, we addressed the issue by introducing a denoising block, highlighted in green in Figure 21, which transforms the log-magnitude spectrogram of $x(t)$ by removing the influence of the additive noise $N_{\text{env}}(f)$. The rationale behind this choice was to develop a universal approach, to be applied independently from the specific feature extraction mechanism or the selected classification model: the log-magnitude spectrogram is a basis foundation not only of the baseline [88], [89], but of the near totality of publications addressing microphone classification.

The denoising procedure which we selected for this purpose is the Denoising CNN (DnCNN) [92]. Given a target original image X and an input image $X_{\text{noise}} = X + N$ corrupted by Gaussian noise N, the Denoising CNN (DnCNN) network is able to compute an estimate $\hat{N}$ of the input noise, and thus an estimate $\hat{X}$ of the original image . In our proposal, the input image X and the Gaussian noise N coincide respectively with the input spectrogram $S(f)$ and environmental noise $N_{\text{env}}(f)$ from eq. (4), as depicted in Figure 22.

Experimental results

The first part of the experiments consisted in training the DnCNN architecture with spectrograms corrupted by white Gaussian noise. For this purpose, we used recordings from the LibriSpeech [93] corpus. We used the *train-clean100* portion of the corpus, corresponding to 100 hours of English speech sampled at 16 kHz and encoded with FLAC. At each epoch, we generated a corrupted version of the input recordings using additive white Gaussian noise with 30 dB Signal to Noise Ratio (SNR), computed the noiseless and noisy spectrograms using 32ms window length and 16ms hop size, and then extracted patches of dimension 256 × 256 to feed the network with. Window length and hop size correspond to the ones in [88], [89], the microphone classification algorithm that we picked as baseline for our experiments. The network converged surprisingly fast, after about 116 iterations.

The modified pipeline including denoising was then tested on the MOBIPHONE dataset [94]. This dataset contains audio recordings from 21 mobile-phones produced by 7 manufacturers, and was specifically devised for evaluating microphone classification. It contains utterances lasting 30 seconds each, spoken by 12 female and 12 male speakers and captured in a silent laboratory environment using a sampling rate of 16 kHz and the GSM-AMR encoding. The final recordings were then distributed as uncompressed WAV files with PCM encoding. Speakers, encoding and devices are thus completely uncorrelated with the ones present in the LibriSpeech corpus.

To obtain our final *noiseless* dataset for training and testing, we split the MOBIPHONE recordings in non-overlapping segments of 4.112 seconds, obtaining 168 *noiseless* examples per class. The same segments were then corrupted using additive white Gaussian noise with 30dB SNR, obtaining 168 *noisy* examples per class. For brevity, we will hereon use $MOBI_{+\infty}$ to denote the *noiseless* examples, and $MOBI_{30}$ to denote the *noisy* ones – i.e., the index denotes the SNR.

To test both the baseline microphone classification approach in [88] and its version modified using our proposed denoising step, we split $MOBI_{+\infty}$ and $MOBI_{30}$ with a ratio of 80% examples for training and 20% for testing, with the extra constraint that the split was identical in both sets. The results of the corresponding evaluation are reported in Table 11.

Table 11. Evaluation of the proposed approach for microphone classification in noisy conditions.

SNR (dB)		Accuracy (%)	
Training	Test	Baseline [88]	Proposed
$+\infty$	$+\infty$	**96.20**	20.73
30	30	21.57	**80.67**

The results show that classification accuracy of the baseline approach in noisy conditions dropped drastically from 96.20% to only 21.57%, as we expected given the noiseless assumptions. After introducing the DnCNN-based denoising step, however, the accuracy raised to back to 80.67%. The increase was possible *without* changing the feature extraction nor the classification stage[11], which is a very encouraging outcome in terms of wide applicability of this approach. The loss in terms of accuracy (-15.53%) is likely to be due to the DnCNN removing also some of the traces left by the microphone frequency response. This loss was also expected, and even if not negligible does not lead the classifier to pure guessing.

At the same time, we saw that the denoising step, if applied to noiseless input content, is degrading the performances of the system from 96.20% to 20.73%. This is due to the fact that we trained the DnCNN only with content degraded by additive white Gaussian noise at a *specific* SNR

[11] The hyperparameters of the SVM involved were also identical

level: The denoiser is not able to distinguish the noise level and is thus always aggressive to the content. As a consequence, for the denoising step to become fully transparent a preliminary noise detection stage should be introduced, and/or the training content of the denoiser should include several levels of SNR, so that the DnCNN does not encounter unknown noise distributions.

Relevant publications

The activities related to microphone classification in noisy conditions lead to a co-supervised M.S. thesis on the topic:

- A. Giganti, "Speaker-independent microphone identification via blind channel estimation in noisy condition," M.S. thesis at Politecnico di Milano (Milano, Italy), 2021 [95].
  Online record: https://www.politesi.polimi.it/handle/10589/179420.

A peer-reviewed submission of the work as conference publication is planned for the near future.

Relevant WP8 Use Cases

This activity relates to manipulated media detection, and thus contributes to the following use-case requirements:

- 1A3 – Synthetic Audio Detection/Verification
- 1A4 – Synthetic Video Detection/Verification
- 2A2 – Factchecking Toolbox
- 3C1 – Just-in-Time Content Verification

4.3 Text-based DeepFake detection

This section presents our approach for the training and evaluation of our DeepFake detection method based on textual information. In Section 4.3.1, we demonstrate our pipeline for the composition of a dataset with DeepFake tweets generated based on three deep generative models trained with data collected from political and public personalities accounts. With the composed dataset, we trained a transformer-based architecture to distinguish DeepFake from original tweets and benchmarked its performance under different evaluation scenarios.

4.3.1 Tweet Generation and DeepFake Detection

The objective of our research is addressing DeepFake text detection. Recent advances in automatic text generation [96]–[99] allow to generate short coherent text that imitates the style of the text on which the models have been trained. Potential misuse of these models includes the spreading of disinformation through quote attribution to prominent political or public personalities. In the context of an increasing political polarisation, this could be detrimental to democracy and carry on actions which may cause public troubles.

Social media platforms have become important sources of information for an increasing number of people around the world. Among these platforms, Twitter allows to spread information quickly around its user community. Twitter posts take the form of short texts, limited to 280 characters. This format is ideal for text generation algorithms and Twitter could therefore be a target for disinformation campaigns implicating such algorithms. In this context, we focus our work on the detection of DeepFake tweets.

Fagni *et al.* [100] investigate the same research question. They collected original tweets from human accounts (accounts the content of which is written by a person) and their fake bot counterparts maintained by people on the Twitter platform (accounts the content of which is written by text generation algorithms). They analysed the performance of several classification models according to the technology used for tweet generation. They show that RoBERTa [101], a pre-trained language model based on the transformer architecture [102] obtains the best performance across all configurations. One limitation of their work is the number of Twitter accounts available for experiments. They retrieved 23 bot accounts and 17 human accounts, thus limiting their evaluation capabilities. Although tweets are different in both train and test datasets, accounts are not unique in either part. This prevents from evaluating the generalisation capabilities of their approach.

In our study, we follow Fagni *et al.* [100] and investigate the capabilities of algorithms to generate accurate and consistent text through a classification task. We address the limitation presented above by creating a new dataset for the task. Our two main contributions are the following:

- We create a new dataset for DeepFake tweet detection which contains around 100 political and public personalities Twitter accounts. We generate their fake counterparts using three deep generative models.
- We investigate the use of several algorithms for DeepFake tweet detection and show that generalisation across accounts remains a difficult task.

This work has been carried out by Babacar Sow during a graduate internship at CEA LIST. He was advised by Adrian Pospecu and Julien Tourille.

Method overview

Dataset – We targeted tweets from 103 accounts including political and public personalities. Accounts include for instance the U.S. Senator Ron Wyden, the former U.S. president Bill Clinton or the TV news anchor Sean Hannity. All these accounts are verified, meaning that the identity of the persons holding these accounts has been checked. We collected tweets using the Twitter API and gathered at most 3,200 tweets per account[12]. We then used 3 distinct models to generate fake tweets based on this corpus.

- GPT-2 [97] is a pretrained generative language model based on the transformer architecture [102]. This model has been trained on the WebText corpus[13], a dataset created for the model which has been scraped over the web (approximately 40 GB of text). We used an open-source implementation available online[14] to finetune the model on the accounts (355M parameters).
- AWD-QRNN [103], [104] is an optimised generative recurrent neural network based on Quasi-Recurrent Neural Networks [105]. We used an open-source implementation available online[15] and used the default parameters recommended by the authors.
- Character Recurrent Neural Networks are a family of generative deep neural networks that work at the character level. We used an open-source implementation available online[16] and used the default parameters recommended by the authors.

Each of these models was finetuned on each account of our dataset in order to generate fake tweets. We did not perform any manual validation of the tweets.

DeepFake Detection – We cast the task as a binary classification task where the objective if to find whether a tweet is fake or original. Based on previous work by Fagni *et al.* [100], we select RoBERTa [101], a transformer-based model for our classifier. We used an open-source implementation available online[17].

We follow Fagni *et al.* [100] for most hyperparameters. We choose to increase the batch size to 64 to accommodate our large volume of data. Our evaluation framework relies on several scenarios.

- In scenario 1, we assess the performance of our classification model to detect fake tweets with several datasets: a mix of fake tweets from the three generation algorithms and three datasets generated with only one model.
- In scenario 2, we train our classifier on tweets generated with GPT 2 and assess its performance on datasets composed of tweets generated by the two others technologies.
- In scenario 3, we trained our classification model on tweets generated by GPT-2 and assess its performance on the test dataset gathered by Fagni *et al.* [100]. We perform three experiments, one on the full dataset, one on the political accounts and one on the non-political accounts.

Experimental results

Classification scores for scenario 1 are presented in Table 12. When trained and tested on the tweets generated by the AWD-QRNN model, the classifier perform almost perfectly with an accuracy score of 0.996 and a F1-score of 0.996. The performance decreases with the two other models. Tweets generated by GPT-2 are the hardest to detect. In this configuration, the classifier

[12] Upper-bound limit set by the Twitter API

[13] The dataset has not been released

[14] https://github.com/minimaxir/gpt-2-simple

[15] https://github.com/fastai/fastai/tree/master/

[16] https://github.com/Jmete/deepThorin

[17] https://github.com/ThilinaRajapakse/simpletransformers

Dataset	Accuracy	F1-score
AWD-QRNN	0.996	0.996
Char-LSTM	0.910	0.915
GPT 2	0.792	0.807
Mixed	0.889	0.897

Table 12. Performance for scenario 1 – This table presents the performance of our classifier on four datasets, one for each text generation algorithm (AWD-QRNN, Char-LSTM and GPT 2) and one composed of a mixed set of tweets generated by the 3 models (Mixed).

obtains an accuracy score of 0.792, well below the score obtained with tweets generated by the Char-LSTM algorithm (accuracy score of 0.910). When using a mixed dataset composed of tweets generated by the three models, the performance of the classifier drops significantly in comparison to the best score (0.889).

Test dataset	Accuracy	F1-score
AWD-QRNN	0.891	0.888
Char-LSTM	0.815	0.825
Mixed	0.851	0.854

Table 13. Performance for scenario 2 – This table presents the performance of our classifier trained on tweets generated by GPT 2 and tested on 3 datasets, one for each text generation algorithm besides GPT 2 (AWD-QRNN and Char-LSTM) and one composed of a mixed set of tweets generated by the 3 models (Mixed).

Performance for scenario 2 is presented in Table 13. When trained on tweets generated by GPT-2, the classifier performed well on other datasets (AWD-QRNN and Char-LSTM) with F-scores of 0.888 and 0.825 respectively. However, these results highlight the difficulty to transfer the knowledge acquired during training on one type of tweets (in this case, GPT-2) to another. The classifier seems to learn specific features depending to the technology. This behaviour need to be investigated in future work.

Test dataset	Accuracy	F1-score
Full test	0.614	0.492
Political accounts	0.736	0.696
Non-political accounts	0.568	0.422

Table 14. Performance for scenario 3 – This table presents the performance of our classifier trained on tweets generated by GPT 2 and tested on the test dataset of Fagni et al. *[100]. We consider three experiments, one with the full test set (Full test), one with only the political accounts (Political accounts) and one with only the non-political accounts (Non-political accounts)*

Performance for our third and final scenario is reported in Table 14. For all three cases, the classifier has been trained on tweets generated with GPT-2. We observe a performance discrepancy depending on the dataset used for test. When applied on the full test set of Fagni *et al.* [100], our classifier obtains a f1-score of 0.492, well below the score reported in the original paper (0.896). Interestingly, the performance is better for accounts that are considered as political than those who are considered as non-political (0.696 vs. 0.422). This discrepancy suggests that the classifier is also dependent on the domain of the tweets presented during training. Further investigation is needed to better understand this dependency.

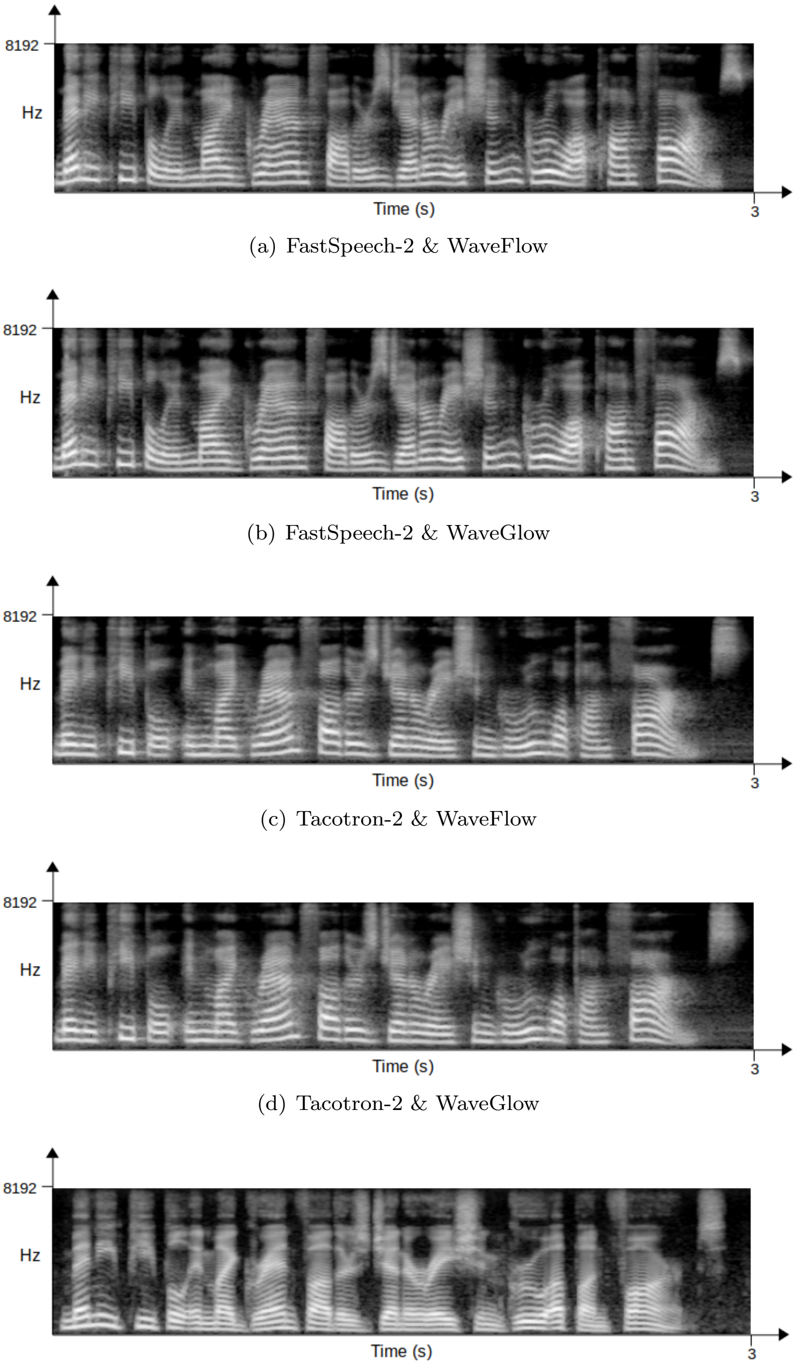
Final Remarks & Perspectives

The research presented in this section is still on-going and several aspects need to be complemented and improved. We observe a large discrepancy between the score obtained by Fagni *et al.* [100] and our classification model trained on tweets generated with GPT-2. This discrepancy could be the result of a combination of several factors. The quality of our generated algorithm could be not optimal. In our experiments, we used the 355M parameter version of the model and followed recommended settings for learning and generation. However, there are more powerful versions of the model (1500M parameters) that could be used for the task. Moreover, a grid-search or a Bayesian optimisation of the hyper-parameters could improve the resulting quality of the generated text. Another aspect to consider when investigating the performance difference is the training data. Fagni *et al.* [100] used the same accounts in both training and test datasets. The task could be considered as easier in this situation. Finally, as shown in the experiments, the performance decreases when our classifier is trained on tweets generated by GPT-2 and applied to tweets generated by another model, thus partially explaining the score reported in Table 14 as the test set is composed of tweets generated by several models.

Another interesting research avenue is to better understand the influence of domain on the classifier quality. As shown in Table 14, domain variation seems to have an influence on the capacity of the model to detect DeepFake tweets.

Relevant WP8 Use Cases

CEA will provide a tool for deep fake tweet detection to the WP8 Use Cases, contributing to the user stories 1A1 (Synthetic Text Detection/Verification), 2A2 (Factchecking Toolbox), and 3C1 (Just-in-Time Content Verification).



(a) FastSpeech-2 & WaveFlow

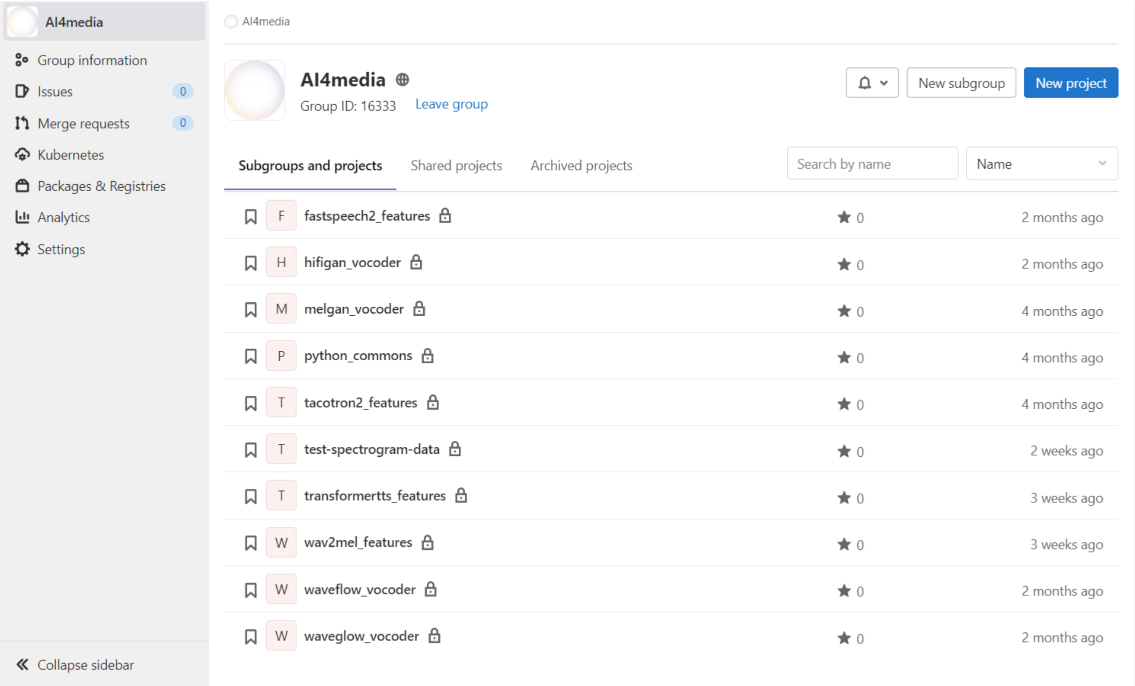
(b) FastSpeech-2 & WaveGlow

(c) Tacotron-2 & WaveFlow

(d) Tacotron-2 & WaveGlow

(e) googleTTS

Figure 17. Mel Spectrograms extracted from the audio outputs of different combinations of Synthetic Speech Synthesis pipelines. The results are based on a text sample input. (a)(b) FastSpeech-2 was used as a feature extractor in the pipeline process.(c)(d) Tacotron-2 was used as a feature extractor in the pipeline process. (e) Audio was produced by Google cloud Text-To-Speech service.



AI4media

Group ID: 16333 Leave group

F fastspeech2_features	★ 0	2 months ago
H hifigan_vocoder	★ 0	2 months ago
M melgan_vocoder	★ 0	4 months ago
P python_commons	★ 0	4 months ago
T tacotron2_features	★ 0	4 months ago
T test-spectrogram-data	★ 0	2 weeks ago
T transformertts_features	★ 0	3 weeks ago
W wav2mel_features	★ 0	3 weeks ago
W waveflow_vocoder	★ 0	2 months ago
W waveglow_vocoder	★ 0	2 months ago

Figure 18. T6.2: Common repositories for synthetic speech generation.

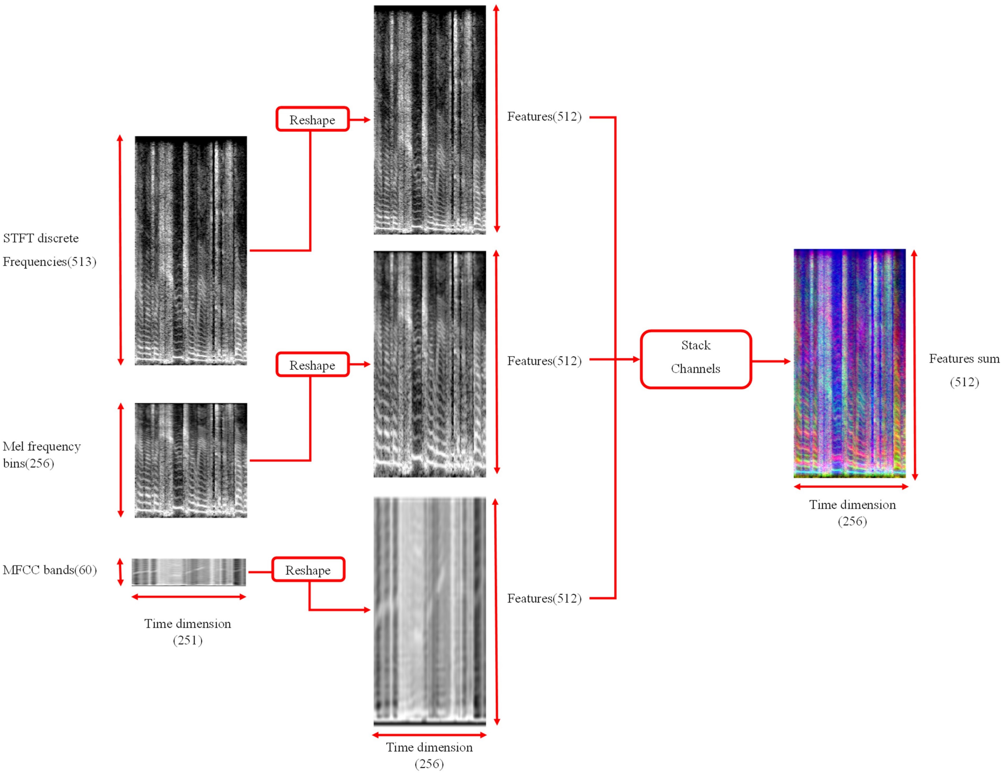


Figure 19. STFT, MFCC, Mel Spectrograms and the multi-channel stacked image.

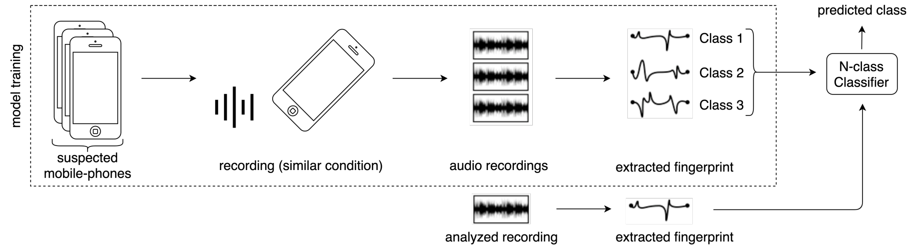


Figure 20. High level schema for closed-set microphone classification.

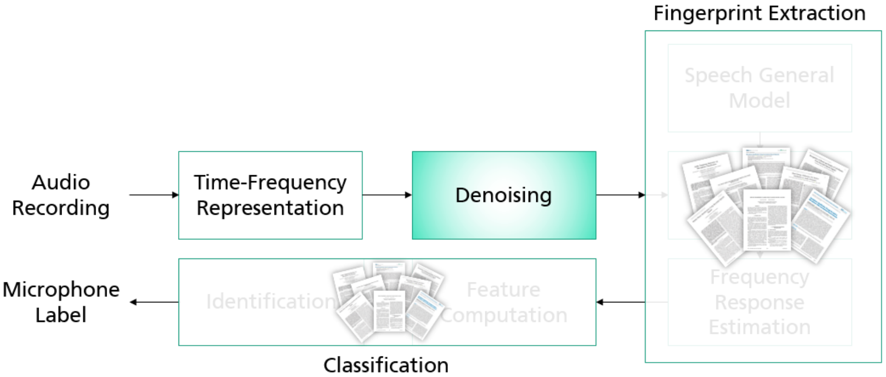


Figure 21. Schema of the proposed approach for microphone classification in noisy conditions.

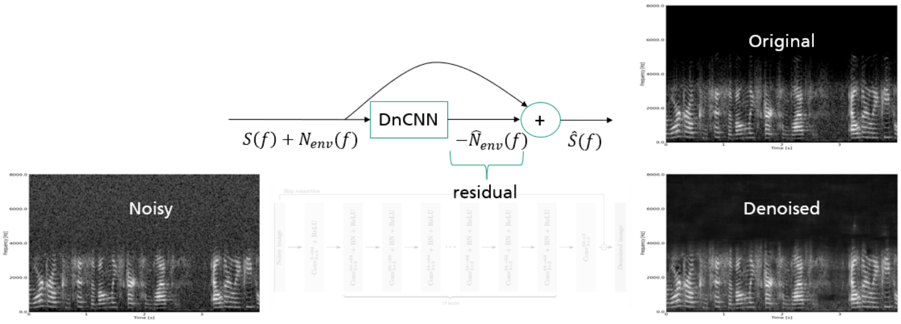


Figure 22. DnCNN application to audio spectrograms.

5 Hybrid, privacy-enhanced recommendation (T6.3)

Contributing partners: FhG-IDMT, UPB

This task provides a privacy aware recommender system for AI4Media. Recommender systems are a powerful tool and shaped huge parts of what we now perceive as the Internet and with it large part of our society. Recommender systems come in all shapes and variations, but there is a common denominator: providing "items" to "users" in a way that a certain utility metric is met ("The user finds the item useful"). This very simple definition already points at two major problems that showed in the last decades of using recommender systems:

1. Measuring "usefulness" or "utility" for a user is hard. Optimizing recommender systems mostly on recorded user behavior led to an effect now prominently called "Filter bubbles". While this might be tolerable for "simple" things like music recommendations, in social networks - also huge recommender systems - this led downright to societal problems, as people were unable to look outside their recommended filter bubble. In AI4Media, we want to look at alternatives.

2. The second problem we want to address in AI4Media comes with the fact that a useful recommender system must know something about its users. Either to personalize recommendations or at least for an evaluation. Of course, the more data available, the more information there is to calculate recommendations from - so the more data, the better. Of course, this goes against the goal of user privacy and possibly regulations like the GDPR. In AI4Media, we want to research if it is possible to apply privacy preserving technologies to user data in a way that the recommendation quality is not degraded significantly. We assume that, while having more data - in theory leads to better recommendations - in practice, also anonymized, pseudonymized and otherwise modified user and usage data can lead to a good enough quality.

To try all out all the things it is important to concentrate on a single use case, as every recommendation scenario is different and it would be unfeasible to create several independent recommender systems in the project. However, the research performed on one recommender will also provide plenty of information for the design of other (future) recommender systems. The use case mapping made in WP8 showed that the Smart News Assistant from UC#2 is a viable option to showcase solutions for the two problems mentioned above (see Deliverable D8.1 - *Use Case Definitions and Requirements* for a comprehensive description of this use case).

AI4Media will provide a lot of components that will be useful to analyse media data and anonymize user data in a way that the above scenario can be implemented. Also for this reason, the technical implementation of this task hasn't started yet, to give the relevant contributing tasks a head start and avoid wasting resources. The actual implementation work will start in 2022.

AI4Media itself is dedicated to research on many aspects that are useful for the recommendation case sketched above. So far we identified the following tasks shown in table 15 as potential integration points into the recommender system.

Of course, this list is not exhaustive and we assume that if the AI4EU AI-on-demand platform gets traction we can also use components from there. On the other hand, this shows how tightly integrated this task is with the whole AI4Media project. It certainly will not be possible to deeply integrate all work of all the mentioned tasks inside the deliverable, but a major goal of the recommender system architecture is to make it easy to add AI modules by some sort of plug-in system. This way, we can easily get started with the basic functionality, but will not be stuck with the developed implementation when the available state of the art advances and new components with the right TRL become available.

Task ID	Description	Comment
3.7	Learning to count	Helper for text classification
4.3	Novel methods for explainable and interpretable AI	Avoiding “black box” recommender systems is crucial for the proposed use case
4.4	Privacy- and security-enhanced federated learning approaches	Federated Learning scenarios might be used to analyze recommended media items, proposed privacy techniques can also be useful in a non-federated setting.
4.5	Methods for detection and mitigation of bias affecting fairness in recommender systems	Crucial task for implementing an “Anti-Filter-Bubble” recommender system.
5.1	Media analysis and summarization	Content based aspects of the recommender system.
5.4	Language analysis in Media	Baselines for (news) text analysis
6.1	Policy recommendations for content moderation	Input on legal aspects
6.2	Manipulation and synthetic content detection in multimedia	Additional information on user generated content.
6.4	AI for Healthier Political Debate	Content and user analysis
6.5	Detection of perceptions of hyper-local news	Helpful social media analysis
6.5	Measuring and Predicting User Perception of Social Media	Critical part of evaluation

Table 15. Potential synergies with other AI4Media tasks.

Relevant WP8 use cases

As described above, the task mainly tries to address use case 2, specifically 2B, 2B2, 2B3, 2B4 and 2C1.

6 AI for Healthier Political Debate (T6.4)

Contributing partners: BSC, AUTH, CEA, UvA, UNIFI

6.1 Neural knowledge transfer for improved sentiment analysis in texts with figurative language

Sentiment analysis in texts, also known as opinion mining, is a significant Natural Language Processing (NLP) task, with many applications in automated social media monitoring, customer feedback processing, e-mail scanning, etc. Despite recent progress due to advances in Deep Neural Networks (DNNs), texts containing figurative language (e.g., sarcasm, irony, metaphors) still pose a challenge to existing methods due to the semantic ambiguities they entail. In this work, a novel setup of neural knowledge transfer is proposed for DNN-based sentiment analysis of figurative texts. It is employed for distilling knowledge from a pretrained binary recognizer of figurative language into a multiclass sentiment classifier, while the latter is being trained under a multitask setting. Thus, hints about figurativeness implicitly help resolve semantic ambiguities.

The proposed method exploits knowledge distillation [106] for increasing DNN-based sentiment analysis accuracy on texts with FL that employ sarcasm, irony and/or metaphor. Thus, during training, a teacher-student architecture is utilized to enrich the student model with the knowledge of a pretrained FL recognizer. The latter one is a binary classifier, while the former one is a multiclass classifier tasked with identifying sentiment in input texts. Thus, due to the nature of the teacher and the different task it solves compared to the student, an atypical kind of distillation is proposed.

Below, all neural models are assumed to be trained with error back-propagation and a variant of stochastic gradient descent. Let us also assume that a DNN-based binary text classifier F has been pretrained under a regular supervised setting on a database containing two classes: “figurative”, “literal/non-figurative”. Since it is common for binary neural classifiers to end with a single sigmoidal neuron, we assume this is the case for F. Thus, a real-valued scalar output of 0/1 corresponds to figurative/literal class prediction, respectively, while a typical output $F(\mathbf{x})$ for a respective input data point $\mathbf{x}$ would actually lie in the interval $[0, 1]$.

The student S is the neural model we actually want to optimize; on a different, sentiment-annotated dataset. Without loss of generality, we assume that it is being trained under a supervised multiclass text classification setting. Typically, $N \geq 3$ classes are employed for the sentiment analysis/opinion mining task (“positive”, “neutral”, “negative”, etc.) and a final softmax activation layer used for deriving the class prediction. S is trained by a regular, suitable loss function $\mathcal{L}_S$, such as Cross-Entropy (CE).

The proposed method consists in training S with the following multitask loss function:

$$\mathcal{L}_M = \mathcal{L}_S + \alpha \mathcal{L}_D, \tag{5}$$

where $\mathcal{L}_D$ is being computed at each iteration by exploiting the pretrained F. In essence, $\mathcal{L}_D$ distills the knowledge of F concerning the current training data point $\mathbf{x}$, i.e., $F(\mathbf{x})$. As noted in [107] for the deep linear scenario, sigmoidal output activation for the binary classification case is equivalent to *soft labels* typically employed for softmax-based multiclass distillation [106]. To compute this loss term, a parallel output layer S_b serving as an auxiliary binary classification head is architecturally plugged onto the penultimate layer of S, while $F(\mathbf{x})$ serves as real-valued/continuous substitute “ground-truth” for $\mathcal{L}_D$. To avoid confusion, the normal softmax-based multiclass classification head of S is denoted below by S_m. Thus, assuming N sentiment classes, S_m/S_b is an output neural layer consisting of $N/1$ neuron(s), respectively.

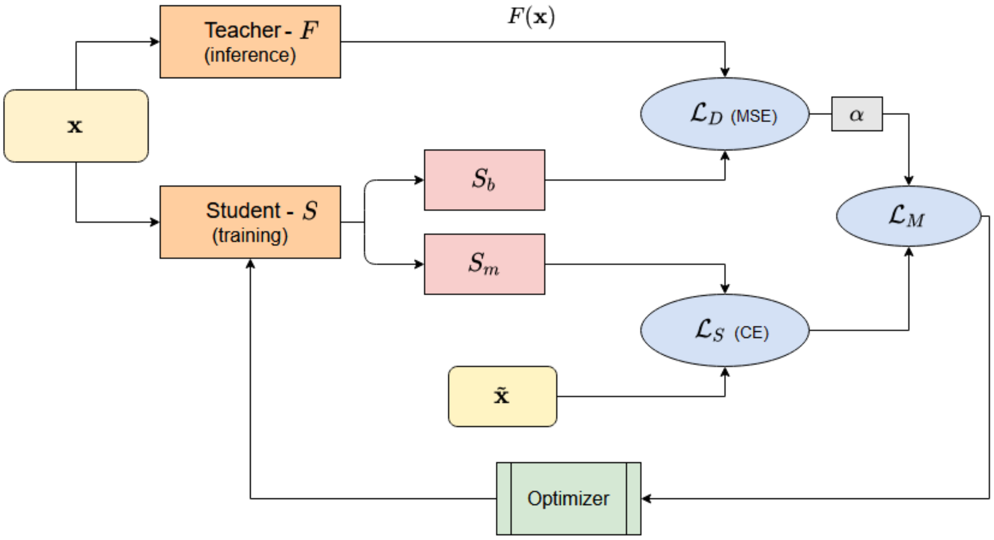


Figure 23. The proposed teacher-student training architecture.

By employing the Mean Squared Error cost (MSE) for the proposed term $\mathcal{L}_D$, the complete multitask loss function is:

$$\mathcal{L}_M = \mathcal{L}_S \left(S_m(\mathbf{x}), \tilde{\mathbf{x}} \right) + \alpha \left(S_b(\mathbf{x}) - F(\mathbf{x}) \right)^2 , \tag{6}$$

where $\tilde{\mathbf{x}}$ is the actual, *sentiment* ground-truth class label corresponding to $\mathbf{x}$, in the context of multiclass classification. Notably, no *figurativeness* ground-truth label is exploited or required to exist for $\mathbf{x}$.

An overview of the proposed method is depicted in Figure 23. Importantly, no actual/real ground-truth annotation concerning the presence or type of FL is required or exploited while training S for sentiment analysis. Of course, after S has been fully trained, both the entire F model and the auxiliary output layer/binary classification head S_b can be safely discarded.

The underlying intuition behind the proposed multitask loss function is the conjecture that dark knowledge concerning the degree of figurativeness of an input text should aid a sentiment classifier in resolving ambiguities about the expressed sentiment, that arise due to sarcastic, metaphorical or ironical language. The proposed distillation loss term should have the effect of tuning the multiclass sentiment classifier towards identifying and overcoming such ambiguities. The FL recognizer F was selected to be a binary classifier in order to maximize its inference-stage success rate in this auxiliary task, by making the classification problem as easy as possible.

Experimental results

The neural architecture ROB-RCNN from [108] was recreated and adopted for the base sentiment analysis student model S. This neural architecture utilizes a pretrained RoBERTa language model [109], combined with an RCNN [110], in order to efficiently capture contextual text information when representing each word. The final prediction is the output of a softmax layer.

The CNN/Bi-LSTM neural architecture OSLCfit [120] was pretrained for FL recognition, following the training process prescribed in [120], and then adopted as the binary classification teacher model F. Its input text representations are derived by using 200-dimensional embeddings from

Table 16. Evaluation results on the S15-T11 dataset. Higher/lower is better for the COS/MSE metric, respectively. Best results are in bold.

Method	***COS***	***MSE***
ELMo [111]	0.71	3.61
USE [112]	0.71	3.17
NBSVM [113]	0.69	3.23
FastText [114]	0.72	2.99
XLnet [115]	0.76	1.84
BERT-Cased [116]	0.72	1.97
BERT-Uncased [116]	0.79	1.54
RoBERTa [109]	0.78	1.55
UPF [117]	0.71	2.46
ClaC [118]	0.76	2.12
DESC [119]	0.82	2.48
ROB-RCNN [108]	0.82	1.92
ROB-RCNN + Proposed ($\mathcal{L}_{\mathcal{D}}$)	**0.85**	**1.50**

a pretrained GloVe model [121]. This teacher model was pretrained on the annotated dataset from [122], which contains 81,4K tweets grouped under 4 different class labels ("sarcasm", "irony", "figurative" and "regular"). The first three classes were combined in a general "figurative" class, in order to train F as a binary figurative text classifier. The student S was trained using Adam optimization and Cross Entropy (CE) as the main multiclass classification student loss function $\mathcal{L}_S$.

The S15-T11 dataset [123] was used for evaluating the proposed method and comparing it against competing approaches. It contains 8,000/4,000 tweets for training/test, respectively, including tweets with ironic, sarcastic and metaphorical language. The 12,000 data points are grouped under 11 classes annotated with integers in an 11-point scale, ranging from -5 to +5, that denote the polarity of each tweet, from "very negative" to "very positive". Since it is a sentiment analysis dataset, it *does not* contain ground-truth annotations/labels concerning the presence or type of FL.

Two common evaluation metrics were employed: cosine similarity (COS, higher is better) and Mean Squared Error (MSE, lower is better). Assuming a test set of T data points, both are computed by comparing two T-dimensional integer vectors, respectively containing the predicted and the ground-truth class labels. The proposed method's implementation is in fact ROB-RCNN [108] augmented with $\mathcal{L}_{\mathcal{D}}$ during training, while the baseline that we improve upon is ROB-RCNN trained with simple $\mathcal{L}_S$, instead of the proposed Eq. (6). Optimal hyperparameters were adopted from [108], while 5-fold cross-validation resulted in best $\alpha = 0.5$. Test-phase evaluation results are presented in Table 16, including the accuracy achieved by several competing methods. All reported figures are lifted from [108], except the ones for ROB-RCNN. The latter method was recreated, trained and evaluated ab initio by us, following strictly all implementation minutiae and hyperparameter values detailed in [108].

Overall, the proposed method implementation ROB-RCNN + $\mathcal{L}_{\mathcal{D}}$ achieves state-of-the-art performance in both metrics, thus confirming the validity of our underlying intuition. Remarkably, compared to DESC, it manages to decrease MSE from 2.48 to 1.50, while simultaneously increasing

COS from 0.82 to 0.85. In contrast, baseline ROB-RCNN achieves MSE improvements over DESC, without gains in COS performance.

Relevant publications

- D. Karamouzas, I. Mademlis, I. Pitas, ”Neural knowledge transfer for improved sentiment analysis in texts with figurative language”, technical report, submitted as conference paper.

Relevant WP8 Use Cases

1C1 (Single Twitter Account Content Analysis). The presented method improves upon the state-of-the-art for semantic analysis of individual tweets. Moreover, figurative language is highly used in social media.

6.2 Public opinion monitoring via semantic analysis of tweets

Twitter is gaining increasing popularity especially within the field of politics. We usually observe increased traffic in this platform during an event like parliamentary/presidential/national elections. To monitor the views of people relating to a specific topic, opinion mining is often performed through sentiment analysis of tweets. Monitoring the public opinion is a very powerful tool as it can be used to predict future outcomes of events or adopt profitable strategies. With that notion we create a novel public opinion monitoring mechanism that consists of two basic tools: A descriptor getter tool and a time series forecasting tool. The former outputs a sentiment descriptor/vector for each tweet containing four values regarding the semantics of the tweet. These values are connected to positivity, offensiveness, bias, and figurative language. The latter tries to predict the future descriptor based on past observations. Both tools are implemented using deep learning models. The novelty of our mechanism lies in the combination of multi-sentiment analysis with time series forecasting, which is something no previous study incorporates.

The proposed mechanism for monitoring and predicting the public opinion consists of two tools. The first tool is composed by four sentiment classifiers and we use it to get the descriptor for each tweet. The obtained descriptors are aggregated for each day and the derived time series is fed to the second tool of our mechanism. That tool would be a forecasting model trained to predict the next week’s descriptors. An overview of the proposed mechanism is depicted in Figure 24.

Two different deep neural architectures were used. We train the hybrid CNN-LSTM architecture from [120] to create three separate classifiers that detect offensiveness, bias and figurative language in tweets. Three relevant public datasets were used for training, while a pretrained publicly available neural model[18] was used for the polarity feature. All DNNs were trained in a binary classification task.

The descriptors derived through the previous procedure were aggregated per day, in order to create the daily descriptor time series. This is fed into the forecasting model to give us forecasts for a 7-day horizon.

Experimental results

For evaluating timeseries forecasting on the dataset constructed using the proposed descriptor, we adopted the stacked LSTM architecture from a comparative study about forecasting using RNNs [124]. The 2016 USA Presidential Election Tweets dataset[19] was exploited for assessing forecasting

[18] https://github.com/DheerajKumar97/US-2020-Election-Campaign-Youtube-Comments-Sentiment-Analysis-RNN-Bidirect--lstm-

[19] https://www.kaggle.com/paulrohan2020/2016-usa-presidential-election-tweets61m-rows

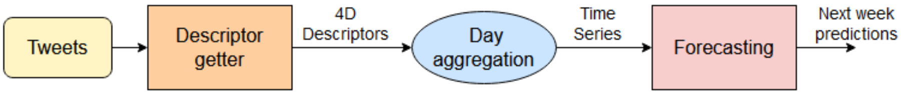


Figure 24. The proposed public opinion monitoring mechanism.

performance. The dataset contains 61 million rows in total from which we keep approximately 32 million after cleaning process. The time span of the tweets is from 2016-08-30 to 2017-02-28 with 20 days missing so having a total of 163 days.

We wanted to monitor public opinion separately for each of the two parties (Democrats and Republicans). To accomplish that, we used the keywords "Clinton", "Obama" and "Trump" to categorize the tweets as the ones referring to Democrats and those referring to Republicans. Therefore, we obtained the descriptors of tweets that represent the public opinion towards the two most popular parties. To create the time series, we applied day aggregation in three different ways. Mean, Median and Trimmed mean were used to obtain a single descriptor value for each day. So we ended up with three datasets for each party, that contain the daily descriptor from 2016-08-30 to 2017-02-28. Of course, each dataset contains four unidimensional time series: one for each dimension of the descriptor. The created time series are used to train our forecasting model.

Table 17. Forecasting results on time series constructed by applying the proposed semantic tweet descriptor to the US 2016 Presidential Elections Dataset.

Dataset Name	***Mean_SMAPE***	***Median_SMAPE***	***Mean_MASE***	***Median_MASE***
dem_mean	0.1096	0.0563	1.2396	1.0799
dem_med	0.1380	0.0913	1.0620	1.1711
dem_trim	0.1798	0.0676	1.3364	1.0885
rep_mean	0.0529	0.0280	0.7689	0.6937
rep_med	0.0492	0.0330	0.5158	0.5303
rep_trim	0.0737	0.0314	0.6931	0.6549

The symmetric mean absolute percentage error (SMAPE) is the most common performance measure used in many forecasting competitions:

$$SMAPE = \frac{100\%}{H} \sum_{k=1}^{H} \frac{|F_k - Y_k|}{(|Y_k| + |F_k|)/2}, \tag{7}$$

where H, Fk, and Yk indicate the size of the horizon, the forecast of the NN, and the actual forecast, respectively.

Due to lack of interpretability and high skewness of SMAPE [125], the mean absolute scaled error (MASE) metric was also employed [125]. It is defined as follows (for non-seasonal time series like ours):

$$MASE = \frac{\frac{1}{H}\sum_{k=1}^{H}|F_k - Y_k|}{\frac{1}{T-1}\sum_{k=2}^{T}|Y_k - Y_{k-1}|}. \tag{8}$$

The MASE is a scale-independent measure, where the numerator is the same as in SMAPE, but normalised by the average in-sample one-step naïve forecast error. A MASE value greater than 1 indicates that the performance of the tested model is worse on average than the naïve benchmark, and a value less than 1 denotes the opposite. Therefore, this error metric provides a direct indication of the performance of the model relative to the naïve benchmark.

The model evaluation of this study is presented using four metrics: the mean SMAPE, median SMAPE, mean MASE, median MASE. Where mean and median are used as aggregators across the four time series that correspond to each dimension of the descriptor. The results are presented in Table 17. Greater values for SMAPE and MASE indicate worse performance.

From SMAPE values we can see that overall our model works very well. For the democrats dataset, the best forecasting results were obtained at the mean aggregation setting while for republicans both mean and median gave good results. From MASE values it becomes clearer that forecasts are more inaccurate for the democrats dataset as the model performs worse than the naïve benchmark.

Relevant publications

- D. Karamouzas, I. Mademlis, I. Pitas, "Public opinion monitoring via semantic analysis of tweets", technical report, under preparation.

Relevant WP8 Use Cases

1B2 (Keyword Analysis/Monitoring of Content in Social Media). The presented method facilitates monitoring of public opinion by semantically analyzing and aggregating Twitter content over time.

6.3 Political tweets in the Greek language

There are currently very limited data resources, in terms of annotated datasets, for semantic analysis of tweets in the Greek language. For example, the only relevant datasets for sentiment classification are [126] and [127]. Thus, during the first 16 months of the project, a large dataset of Greek-language tweets with political content was collected by partner AUTH in the context of T6.4.

Approximately 900,000 tweets were gathered, based on their hashtags, covering a period from January 2014 up to May 2021. More than 40 different hashtags were employed for selecting relevant tweets, mostly composed of names of mainstream political parties, party leaders or major political events (e.g., the referendum of July 2015).

A small subset of approximately 4,000 tweets, randomly selected from the entire dataset, was manually annotated with ground-truth labels for the four semantic features mentioned in 6.2 (polarity, bias, offensiveness, figurativeness). Neural representations for each tweet were then derived using the fastText architecture [128], pretrained in Greek[20].

Experimental evaluation for tweet classification in this subset, separately along each of the four features, is currently underway and will be reported in the next version of this deliverable.

Relevant publications

- I. Koroni, I. Mademlis, I. Pitas, "A dataset of politically charged tweets in Greek language", technical report, under preparation.

Relevant WP8 Use Cases

1C1 (Single Twitter Account Content Analysis). A large-scale Greek-language tweet dataset facilitates research on social media content analysis in Greek.

[20] https://fasttext.cc/docs/en/crawl-vectors.html

6.4 Measuring healthiness of online discussions on Twitter using the temporal dynamics of attention data

Potential ways to analyse news events and articles and the related discussions on social media have been a hot topic for a long time. With volume of generated content and the speed of its generation growing higher as time goes by, there is a lot of opportunities to study their laws and patterns. Moreover, whereas the qualitative approach was the traditional way of analyzing news and discussions, their volume offers a chance to move towards quantitative analysis approaches.

Additionally, the amount of misinformation in media grows as well, not to mention various manipulative techniques used by malicious or unscrupulous agents to affect their target audiences in desired ways.

This raises the question of healthy discussions. What discussions can be called healthy? Do they have any specific patterns or properties? Can we define some formal criteria for healthiness of online discussions? Moreover, given the sheer volume of the social media content, is it possible to do the healthiness analysis in quantitative way instead of the traditional qualitative one?

In order to answer these questions we collected a significant volume of tweets and our intention is to try out various techniques to find common patterns between different topics of discussions among them, automatize them, and come up with metrics that can estimate the healthiness of such discussions.

The collection process started in March 2020, in the early days of the COVID-19 pandemic. During the first two days of sudden domestic confinement, we acknowledged the exceptionality of the situation, and started gathering Twitter data, in hopes that such a unique dataset would enable novel research outcomes in the near future. At the time, it was decided to start gathering global (English) and local (Spanish) tweets related to COVID-19, by defining a query made up of a broad range of keywords. Our goal was to obtain a large, representative set of online discussions, which could contain relevant insights on social behaviour during such extreme times. This procedure is the first step shown in the pipeline of Figure 25.

Based on this large data set (approximately 1.2 billion tweets in the period between the 25th of March 2020 and the 24th of March 2021), we isolated topics of conversation to analyse and compare their particularities. By doing so we inherently assume that discussions, and how they unfold, have a high degree of coherency based on the underlying topic. These discussions correspond to a given COVID-19 discussion element (or *topic* for short) and form the basis of our analysis.

The data collection process enabling this work can be divided into 3 stages:

1. Accessing and storing tweets
2. Indexing their text fields for text search with a specialised database
3. Topic extraction

Stages 1 and 2 are illustrated in Figure 25, while stage 3 is illustrated in Figure 27.

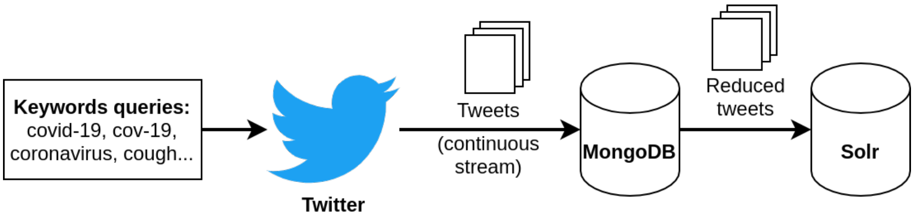


Figure 25. Tweet collection and indexing

For the initial data collection, we used the official Twitter API stream, continuously gathering tweets since the 24th of March, 2020. For that purpose, we composed a list of 50 keywords related to COVID-19 (both in English and Spanish) to be able to monitor both local and global effects of the pandemic.

The standard Twitter API is restricted, limiting the amount of gathered tweets fitting the query. The global distribution of gathered tweets on day-by-day basis is presented on Figure 26, with the average being in the order of 2 to 4 million (additionally offset on several days by hardware issues). While analysing the properties of our collection of tweets, we noticed a sudden change after the 15th of August 2020 (red dashed line on the graph). After this date our system started to receive roughly 50% more tweets from the Twitter API. Upon investigating this phenomenon, we found that around this date (on the 12th of August) Twitter enabled the early access to its API v2[21]. Our hypothesis is that either their policy, its settings, or hardware configuration were modified with the same update. Because of this data irregularity, we decided to process these two time intervals independently:

Period 1: from 25th of March 2020 to 14th of August 2020, mean: 2,592,995, standard deviation: 721,490. Denoted as "¡1¿" after the topic names.

Period 2: from 15th of August to 24th of March 2021, mean: 3,735,678, standard deviation: 667,488. Denoted as subscript "¡2¿" after the topic names.

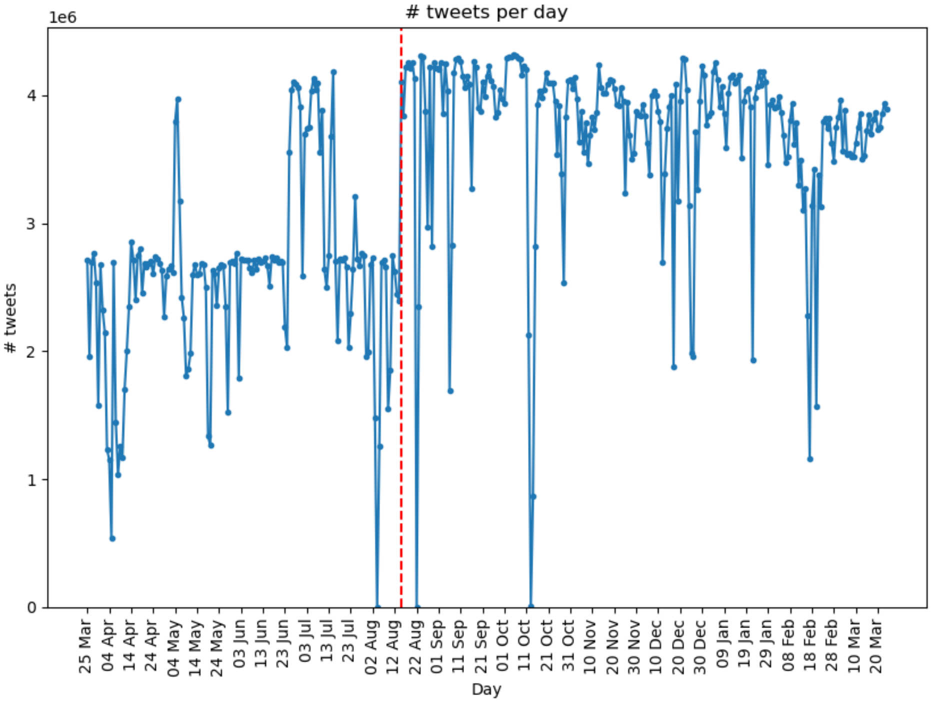


Figure 26. Number of tweets gathered per day in millions in the full collection

[21] https://blog.twitter.com/developer/en_us/topics/tools/2020/introducing_new_twitter_api

Besides the bodies of tweets, we also gathered all available metadata, including the user information and all provided data about tweets that were retweeted, quoted or replied to.

To extract discussion topics, first we came up with a subset of COVID-19-related misinformation news and rumours based on Poynter's database[22]. Using the topics' descriptions, we manually composed text search queries to extract relevant tweets using their most characteristic keywords.

In total, we characterized 78 different misinformation topics within a total timespan of a year, from the 25th of March 2020 to the 24th of March 2021. After splitting the topics into 2 periods and sorting out partial topics that did not see significant activity within the respective periods, we got a total of 129 topics, 61 in the first period and 68 in the second one. For all posterior analysis, we represented each of those topics as an integer vector of either 143 positions (the first period) or 222 positions (the second period), each element of which contains the normalized number of tweets for that topic in the corresponding day (topic distribution vectors). Essentially, these vectors are approximation of attention dynamics for the corresponding topics. Figure 28 contain these topic distribution vectors for the topics from the sample.

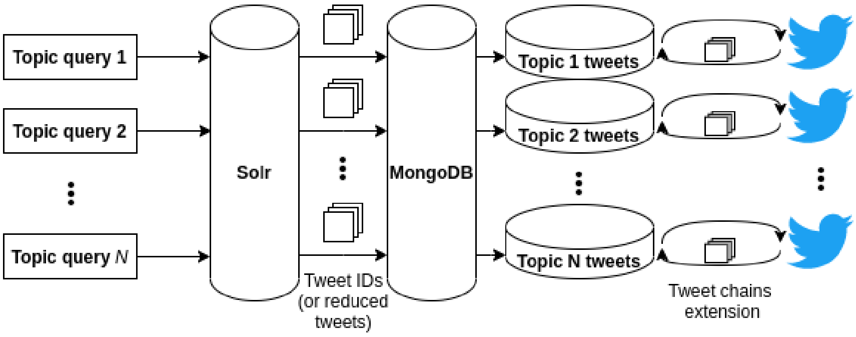


Figure 27. Topic extraction

After defining the topics, one more step of processing was needed to ensure that the discussions are collected as intact as possible. The tweets returned by the query to the Twitter API represent a random sample of the full corresponding set, thus, some quote, reply, and retweet chains were broken by this sampling. Moreover, our hypothesis is that all the tweets in these chains are also relevant to the corresponding discussion topic, but not all of them might have the necessary keywords which could have also led to their exclusion. Therefore, for each tweet, we followed the chains of quotes, replies, and retweets in the direction of the original tweets (as the "parent" tweet IDs are included in the tweet object's metadata) to include these parts of discussions that may have been missing. The whole process is depicted in Figure 27.

[22] https://www.poynter.org/ifcn-covid-19-misinformation/

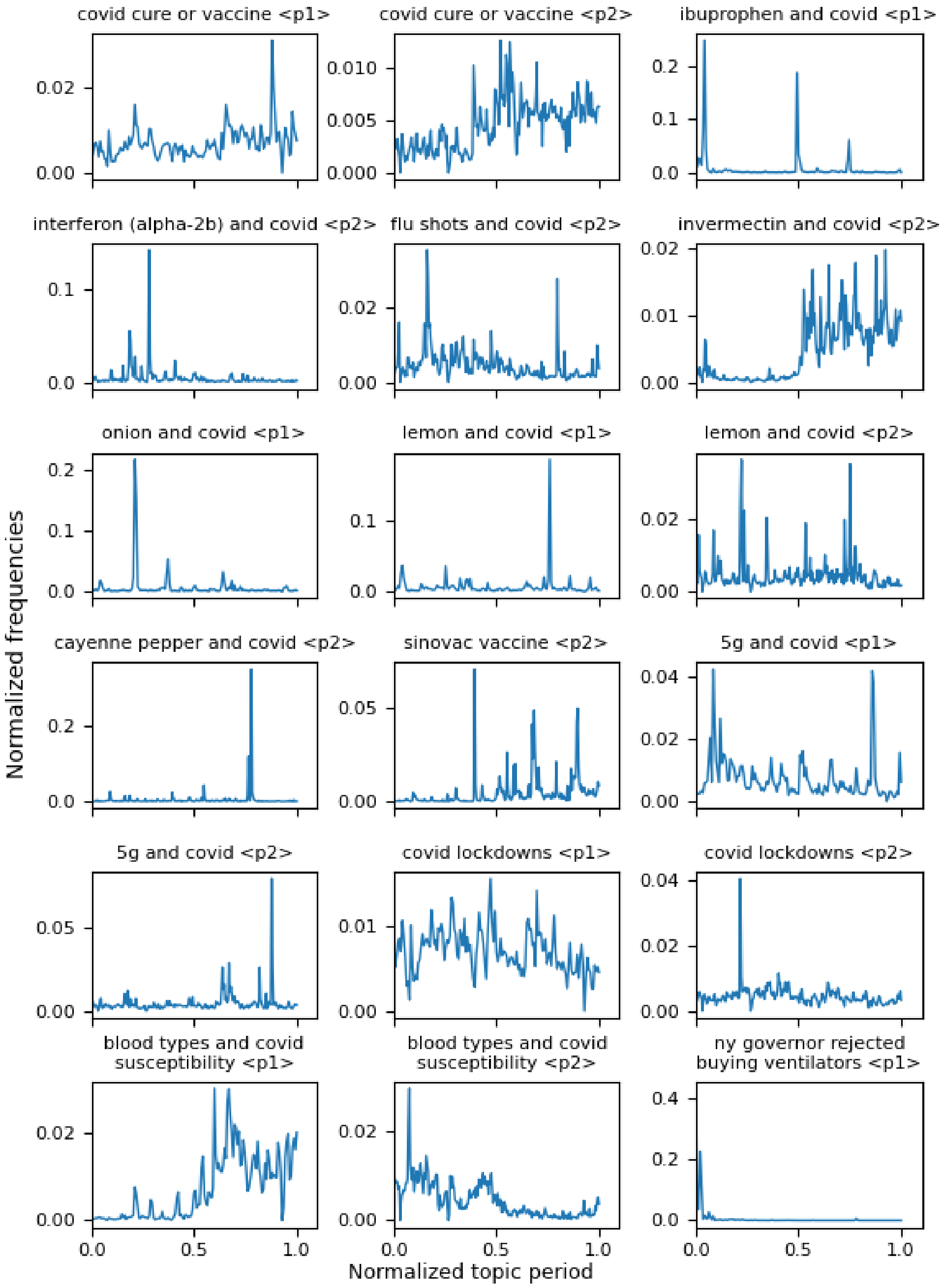


Figure 28. Frequency curves for a sample of topics.

7 Perception of hyper-local news (T6.5)

Contributing partners: IDIAP, CEA, UNIFI

Local news are indispensable sources of information and stories of relevance to individuals and communities [129]. In the European context, these news are produced by hundreds of sources [130]. This task is focused on the analysis of local news and the understanding of their perception both by people and machines. In the reported period, work was done along three lines, all related to health information. The first one is on classifying covid-19-related false information in online news articles. The second one is on building a corpus of local news about covid-19 vaccination across European countries. The third one is on the exploration of online video as another health information source. Each of them is described in a separate subsection.

7.1 Classifying covid-19-related false information in online news articles

In this line of work, we conceptualized local news as coverage of events and stories in local languages, where the covered events have both local and international components, and where the specific local elements have key relevance. The covid-19 pandemic is a good example of such conceptualization, as local news stories occur in specific regional or national contexts, but also have links to the larger international situation.

One contribution in this line of work is the focus on full online articles, as opposed to the work done on short textual social media sources like tweets (the data source in other WP6 tasks). Full online articles represent larger text sources, which can pose challenges to deep learning methods. A second contribution of this work is its interest on non-English news sources. This is important as a good proportion of the top-tier published research on misinformation has been done with English text corpora. Dealing with local European languages is a need for the EU news ecosystem and therefore of relevance for AI4Media. In this case, we focused on news in Spanish, with English as a second reference target. In a larger context, examining local news is important as other authors have argued that understanding the full information ecosystem is key to capture the complexities of issues like misinformation. For instance, based on the results of a large-scale analysis of US media sources, Allen et. al. suggested that ”the origins of public misinformedness and polarization are more likely to lie in the content of ordinary news or the avoidance of news altogether as they are in overt fakery” [131].

Method overview

We located and manually analyzed existing data sources related to covid-19 online articles in Spanish and English, available from both recently published scientific papers and online sources. A subset of these articles had been labeled with one of the many variants used in the literature to flag false information [132]–[135]. A close analysis of this data showed three important challenges. The first one is the large variability of how false information can be labeled: different papers followed different ways of labeling articles as being true or false (or having degrees of falsehood in their statements.) This creates difficulties for aggregating datasets under a common labeling scheme, which is necessary for machine learning approaches. Working with full articles presents difficulties from the perspective of labeling, as long pieces of text could for instance contain one paragraph with false information, while the rest of article could be accurate. In this way, rigorous labeling of full articles in a complex task, and that partly explains the fact that different datasets have different coding systems.

The second challenge were the textual sources themselves. We identified that in some published papers the textual information used for experiments contained elements unrelated to the news articles themselves, likely due to data gathering choices. The third challenge was the amount of available data. After several curation steps, we ended up with a curated dataset of 4,000 articles in Spanish and a second dataset of 8,000 articles in English, labeled with three classes corresponding to: (1) articles with true information, (2) articles containing false information, and (3) articles that debunked false claims by providing true information.

Experimental results

A series of classification models were then trained and tested on this data. Text representations included classic methods like bag-of-words and TF-IDF, as well as word embeddings including Word2Vec and Glove. A variety of classifiers were also assessed, including standard methods like Support Vector Machines, Logistic Regression, Random Forests, and more advanced methods like LSTMs and Transformers (including BERT, Distilbert, RoBERTa, AIBERT, Deberta) . The classification results on this three-class classification task on the Spanish news dataset (true/false/anti-false) showed encouraging performance, with best accuracy over 80% on a balanced three-class setting for all models, and over 90% for the best performing models. As an illustration of all the obtained results, Table 18 summarizes the results obtained with transformer models for the Spanish and the English datasets.

Table 18. Classification results of three-class task on balanced English and balanced Spanish datasets for Transformer based-methods. Rows 1-7 correspond to results for the English dataset. The last row corresponds to results for the Spanish dataset.

Method	**Test accuracy**	**Matthews correlation (MCC)**
Distilbert-base-uncased	82.5	0.73
Distilbert-base-cased	84.0	0.76
Bert-bert-base-cased	84.8	0.77
RoBERTa	86.4	0.79
AIBERT	83.7	0.75
DistilroBERTa	83.7	0.75
Deberta	85.6	0.78
Spanish BERT	97.9	0.96

7.2 Building a corpus of local news about covid-19 vaccination across European countries

The work described in the previous subsection highlighted the need for data of quality about local news across multiple European languages. With this in mind, we set up the goal of collecting a new corpus of local news about covid-19 vaccination published by the main newspapers across European countries. Using the new corpus as input, the aim of this research line is to conduct a series of textual analyses of articles to compare how European newspapers cover the subject of covid-19 vaccination in their local countries. More specifically, we want to analyze the local components associated with each country on common themes such as vaccination attitudes, sentiment towards this theme, and subtopics of relevance, using machine learning and natural language processing techniques. The results of this research line could inform media companies and the public about how topics of common relevance are treated across Europe.

Method overview

We defined a list of over 30 newspapers in five European countries, namely France, Italy, Spain, Switzerland, and UK, and four languages (French, Spanish, Italian, and English.) For each of these newspapers, we had to obtain permission to access their content and curate articles from the newspapers' authorized websites. We asked for permission to each official contact point. At the moment of writing, 15 newspapers (including a combination of national and regional media organizations in each country) gave permission to download and use their content for research, non-commercial purposes. Discussion with several newspapers is currently in progress.

For those newspapers for which permission was obtained, we first verified aspects of technical feasibility regarding data access. Afterwards, we downloaded articles on the covid-19 pandemic and vaccination. The main data we collect includes the title of the article, the main text, and the date of publication. Other metadata is collected if available. This currently corresponds to a corpus of over 40,000 full articles. Table 19 shows a summary of the existing number of articles per country. The dataset might increase in size, if permissions by additional newspapers are granted. This will be a valuable data resource both in focus and coverage. As an illustration of its content, Figure 29 shows the top-20 named entities per country in the dataset, extracted with the spacy library (`https://spacy.io/`), which has models trained in different languages.

Table 19. Status of dataset of news from European newspapers

Country	**# newspapers**	**# articles**
France	2	2395
Italy	2	7547
Spain	6	18969
Switzerland	3	6458
UK	2	5131
Total	15	40500

Once the curation of the current data is completed, the next steps will involve a comparative analysis of local news from different countries occurring in the same period of time. Specific NLP tasks include, but are not limited to: descriptive analysis, sentiment analysis, topic analysis, identification of sub-themes, and temporal analysis.

7.3 Exploring other health information sources: the case of online video

As an exploration of additional sources of local content related to health, we complemented work done in a Swiss national project at the intersection of health psychology and social computing (https://www.idiap.ch/en/scientific-research/projects/HEALTHVLOGGING), and focused on understanding some aspects of how information about health is presented in YouTube from a first-person perspective.

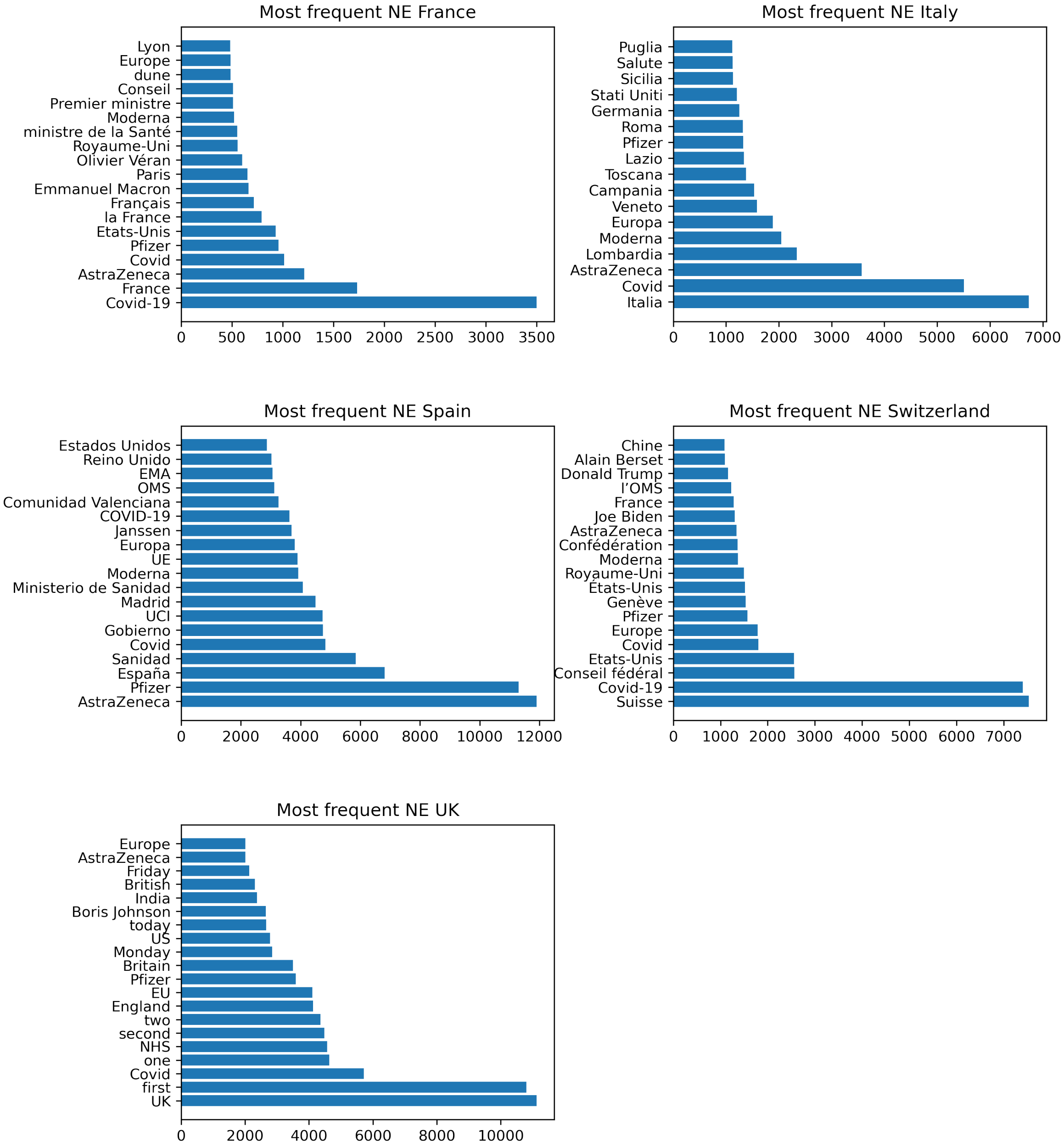


Figure 29. Top named entities per country in European news dataset.

Method overview

We studied a dataset of fifty popular YouTubers who talk about health and wellbeing topics, originally collected and described in [136]. A set of 2,500 videos (mean duration: 10.5 minutes) produced by these users was labeled according to a coding system consisting of six video categories, each of which relates to a health or wellbeing aspect (e.g., food/nutrition or physical activity.) The objectives of the analysis were three-fold. First, we wanted to understand whether such videos span multiple health topics, as is often the case with textual sources. Second, we aimed to understand the kind of linguistic markers and visual representations used to talk about health topics when a first-person perspective is used. Finally, we investigated whether such linguistic and visual markers can be used to automatically identify the main health topics discussed in a given video. Such an approach would be useful to automatically code candidate videos for larger scale studies.

Experimental results

The results for each of these goals can be summarized as follows. For the first goal, we observed that in the studied video sample, health topics indeed occur in conjunction. More specifically, videos often span two or more of the six topics in the coding system, which suggests that a 'soft' way of categorizing health videos is appropriate, i.e., a single video can belong simultaneously to multiple categories.

For the second goal, speech transcriptions were used as input for both classical linguistic marker extraction (LIWC: Linguistic Inquiry and Word Count) and word embedding methods. LIWC in particular allows for interpretable analyses. The results showed trends that are comparable to other sources of first-person content, including text blogs and spoken speech. This included a larger use of first-person speech (singular pronoun 'I') compared to both 'You' and 'We' for five of the six video categories; a larger use of positive emotion terms compared to negative emotion terms for all six video categories; and a larger focus on the present compared to both the past and the future for all video categories. In addition, a visual analysis based on deep-learning-based extraction of semantic scene segments showed that video categories match typical expectations about the scenes where videos are captured, e.g., videos about diet are recorded more frequently in kitchen scenes, and videos about resting are recorded more often in outdoor scenes with trees or plants; this shows the intentionality (but also the imitation practices) of video creation.

Finally, for the third goal, the use of linguistic markers and visual cues in six binary classification tasks (one per video category) shows that performance is promising, more specifically in the range 74-87% in terms of best classification accuracy. Based on the current results, this line of work would have to be investigated in more depth to contextualize it around specific topics of interest (e.g., covid-19 vaccination attitudes) and local contexts (e.g., countries with low vaccination rates).

Relevant publications

There are no published papers from task T6.5 for the reporting period. One paper related to the work in Section 7.3 is currently in preparation, and others are expected for the next reporting period.

Relevant software and/or external resources

The dataset of local news across European newspapers described in Section 7.2 will be a unique dataset when completed. The final version of this dataset will be reported in the next deliverable describing WP6 progress.

Table 20. Binary classification results for six health/wellbeing categories using LIWC linguistic indicators (87 dimensions) and visual scene indicators (150 dimensions). Binary classification tasks (e.g. nutrition video vs. non-nutrition video) were implemented on balanced datasets. N indicates the number of available videos for each class. Accuracy is shown as a 0-1 fraction.

Binary class	**N**	**Scene indicators**	**LIWC indicators**
Nutrition	2120	0.80	0.87
Self-Development	1374	0.69	0.75
Bodycare	1328	0.71	0.78
Physical Activity	878	0.72	0.78
Companionship	370	0.65	0.77
Rest	316	0.68	0.74

Relevant WP8 Use Cases

The work done in Task T6.5 is related to Use Case 2A (Factchecking Toolbox). The analysis of news across multiple European newspapers could enable comparison of news treatment on specific topic and local contextualization. The potential of such analysis in a practical setting would have to be validated after the research work is completed. Importantly, note that automated factchecking by itself is not envisioned as a functionality, and that access to multiple newspapers is assumed to be possible.

8 Measuring and Predicting User Perception of Social Media (T6.6)

Contributing partners: UPB, CERTH, QMUL, UvA

Human perception of multimedia data is complex study domain, filled with an impressive number of positively and negatively correlated concepts [137], that joins the work of researchers from different domains, including social and psychological studies and computer vision. In the context of AI4Media, the main focus of T6.6 is to provide media creators with modern tools and methods that can accurately predict or identify viewer's emotions and perception of created content. During the reported period, work was performed for the following concepts and topics related to human perception of media data: interestingness in section 8.1, memorability in section 8.2, fusion systems for media perception analysis in section 8.3 and affect recognition in sections 8.4 and 8.5.

8.1 Benchmarking and Predicting Media Interestingness in Images and Videos

Visual Interestingness represents one of the key concepts currently being studied related to the human understanding of media content, and to the effect media samples have on human subjects. Being defined as the capacity of "holding or catching attention" in the Oxford Dictionary [138], it has been studied as a psychological factor or component of human behavior and motivation since the 1940s [139]. From an emotional standpoint, interest has been associated with the class of emotions that revolve around comprehension, exploration and desire to learn [140], [141].

In this context, we created a publicly available common evaluation framework for image and video interestingness prediction [142]. The associated dataset, *Interestingness10k*, is composed of 9,831 individual images and over 4 hours of video, interestingness scores generated with a pair-wise annotation protocol summing up to over 1 million pairs of individual annotations, a set pre-computed descriptors for each individual media sample, as well as common data splits and metrics in order to facilitate system development and comparisons. The data were validated during the 2016-2017 edition of the MediaEval Predicting Media Interestingness tasks[23]. We perform an in-depth analysis of several factors that influence the prediction of media interestingness, as well as data and annotation statistics. Starting from the 192 systems validated both during the MediaEval editions of the interestingness task and after the task in state-of-the-art papers, we analyze performance focusing on the best performing features, learning methods and models, method generalization capabilities and system reliability analysis. We further enhance this analysis with the addition of several state-of-the-art general purpose DNNs, as well as a novel DNN-based ensembling method. We propose a set of general observations and suggestions for enhancing system performance based on our analysis.

General observations and suggestions

Based on the analysis of the proposed data and of the 192 systems, we present a set of general observations and suggestions that may help researchers in developing better systems for interestingness prediction:

- As a general trend, when analyzing modalities, visual information expressed as visual features stands out when compared with other modalities. Both deep visual features and traditional

[23] https://multimediaeval.github.io/

descriptors show promising results, on average being the top performers in both image and video prediction;
- Perhaps the most obvious outlier is represented by late fusion systems, which perform significantly better on average than both single-system approaches and early fusion approaches;
- A set of systems showed increased performance when the training data is augmented with external annotations that target concepts related to visual interestingness – like social interestingness and emotional content;
- Data upsampling is another good strategy for increasing system performance;
- Surprisingly, deep neural network approaches may not represent the best performers by themselves. However, on average their performance is very high;
- There seems to be a correlation between the performance of similar (i.e., using the same preprocessing, features, learning method and post-processing) image and video systems. With a Pearson correlation of 0.546 this may seem to indicate that a good place to start working on video interestingness prediction may be image prediction;
- System performance when processing longer (11-12 seconds) videos is better than that for shorter (1-2 seconds) videos even when systems were only trained with short videos.

State-of-the-art deep neural networks

In order to account for the latest developments in deep neural network architectures, we propose evaluating the performance of three popular image and video classification deep neural networks (DNNs), and adapting and fine-tuning them for the Interestingness10k data. Specifically, we utilize the ResNeXt-101-32x48d [143], PNASNet-5 [144], and ResNet-50 [37] networks for image data prediction and the the GSM-InceptionV3 En3 [145], IR-CSN-152 [146], and R(2+1)-18 [147] architectures for video data prediction. These were augmented with best practices as presented in [148].

Performances are presented in Table 21. Performances of the selected DNNs are compared against the best performers both from the MediaEval task (bestME) and from the general literature (bestSoA) and are split according to the data they use (2016 or 2017 version of the dataset). While the results of these networks are promising, surpassing a large part of the 192 submitted systems, they do not have the best performance. This may indicate that training for interestingness prediction may require a more domain-specific approach than just using a state-of-the-art network.

Next, we propose the novel DeepFusion method, a DNN-based ensembling method. To the best of our knowledge, this method represents one of the first attempts at using deep networks as the *ensemble engine* instead of just integrating them as inducers. For this ensembling method we used the 192 submitted systems as inducers, and as we show in Table 21 the results are significantly boosted with the help of DeepFusion. As this method is a general fusion method, that we applied to a large number of different tasks and datasets, we will present it in detail in Section 8.3, together with a larger set of experiments on interestingness, violence, affect recognition and other concepts.

Finally, we use a Grad-CAM architecture [149] to attempt to understand how deep neural networks interpret the visual samples from the Interestingness10k collection. Results are presented in Figure 30. It is interesting to note that, while in many cases the model focuses on the main subject of the image, predominantly more it focuses in regions around the main subject, signaling the importance of context information in predicting interestingness.

Relevant publications

- `M.G. Constantin, L.D. Ştefan, B. Ionescu, N.Q.K. Duong, C.-H. Demarty, M. Sjöberg : "Visual Interestingness Prediction: A Benchmark Framework and`

	Method	2016 data (mAP)	2017 data (mAP@10)
Image	bestME	0.2336	0.1385
	bestSoA	0.2485	0.1560
	FixResNet50 [148]	0.1906	0.1099
	FixPNASNet-5 [148]	0.1981	0.1233
	FixResNeXt-101-32x48d [148]	0.2273	0.1410
	DeepFusion	**0.3459**	**0.2646**
Video	bestME	0.1815	0.0827
	bestSoA	0.1815	0.0930
	IR-CSN-152 [146]	0.1577	0.0629
	R(2+1)-18 [147]	0.1579	0.0644
	GSM-InceptionV3-En3 [145]	0.1738	0.0821
	DeepFusion	**0.2985**	**0.3202**

Table 21. Performance of popular deep neural network architectures when trained on the Interestingness10k data, compared with the best results recorded so far on the Interestingness10k data.

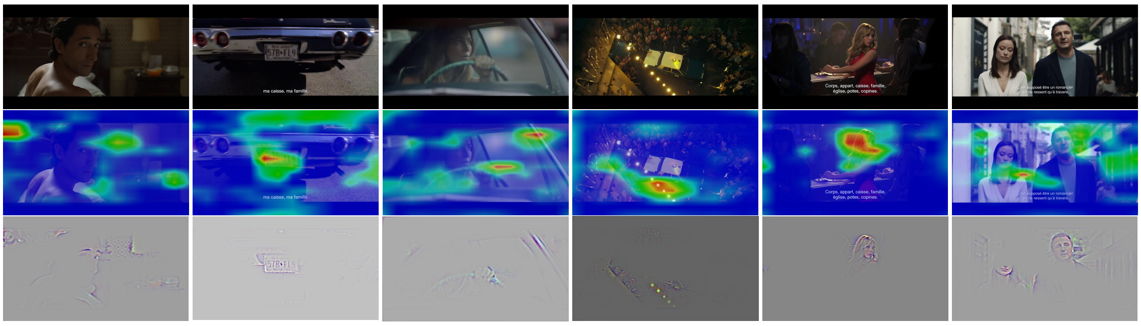


Figure 30. Grad-CAM analysis of the network interpretation in the case of images predicted as interesting: original samples are displayed on the top row, class-discriminative regions on the middle row and dominant features on the bottom row.

`Literature Review". International Journal of Computer Vision, February 2021.` [142].
Zenodo record: `https://zenodo.org/record/5006039`.

Relevant software and/or external resources

- The Interestingness10k dataset, data splits, pre-computed features and metrics are found at: `https://www.interdigital.com/data_sets/interestingness-dataset`.

Relevant WP8 Use Cases

3C12 (Multimodal sentiment analysis). Interestingness as a measure of the ability of multimedia items to attract and hold viewer attention is a concept that relates to the global user sentiment analysis.

8.2 Predicting Video Memorability

In recent developments, Vision Transformers have shown their usefulness for image processing, surpassing convolutional approaches in image recognition tasks [150]. To the best of our knowledge, this approach is relatively untested in the domain of human perception in general and media memorability in particular. This is perhaps to be expected, as the rise of Vision Transformers is in itself a novelty at this point in time. The proposed method for video memorability prediction relies on the use of Vision Transformer networks for feature extraction, a dense network ending for sample regression and a frame filtering method that attempts to feed only the most representative frames to the ViT extractor. We call these representative frames ”Memorable Moments”. This architecture is presented in Figure 31.

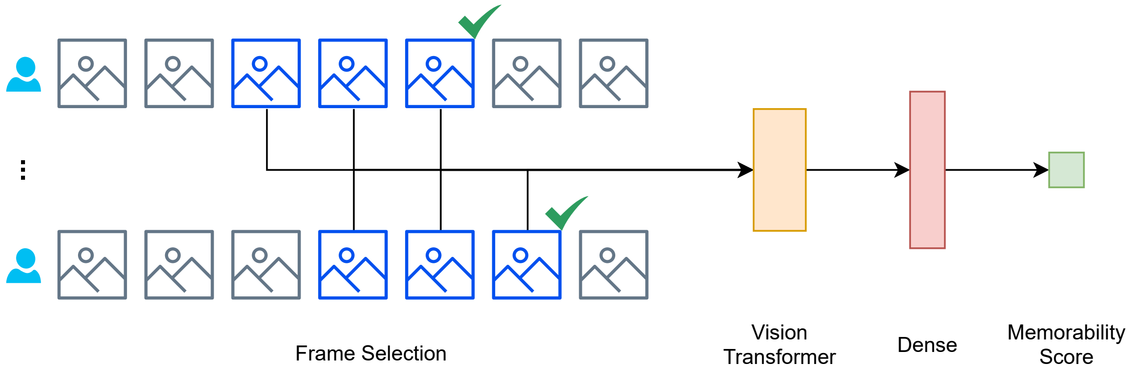


Figure 31. The diagram of the proposed frame filtering solution in the training stage. The Frame Selection phase uses the most representative images from the entire set of frames in a video that create the Memorable Moments, that are then processed by a Vision Transformer architecture and processed by a Dense MLP head in order to obtain the final Memorability Score. Annotator picks on the training set for the Memorable Moments are presented with a green tick mark.

Memorable moments

We base our frame filtering technique on the notion that not all frames are equal when attempting to establish the attributes of a longer video sequence. In our proposed method, we will rely, in the training phase, on the annotations provided with the memorability datasets, and use them to select the frames that best characterize the film in terms of memorability. We name these frames ”Memorable moments,” and while they may not accurately capture the exact moment or process of human memory retrieval, we believe they are a better method than simply analyzing the full film. Several parameters can influence the way frames are selected, having to do with human reaction times and the number of significant frames. Therefore we propose the following parameters for setup:

- Given rt, the time of response from annotators calculated from the start of the film, we subtract the following values in order to take into account the delay between annotator memory recall moment and button press in the annotation tool: $rt' = rt - d$, with $d \in 500, 1000, 1500$ milliseconds;
- We take a variable number of frames before the frame corresponding to the rt' time, namely $15, 30, 60$ frames.

We theorize that using only Memorable moments when training the system will improve the overall results, when compared with a normal training approach, where the entire video is taken into account.

			R1	R2	MediaEval top
Subtask	Dataset	Annotations	Spearman	Spearman	Spearman
Subtask 1	TRECVid	short-raw	0.293	**0.297**	0.297
		short-normalized	0.26	0.251	**0.293**
		long	0.079	0.097	**0.125**
	Memento10K	short-raw	0.407	0.648	**0.658**
		short-normalized	0.641	0.648	**0.658**
Subtask2	TRECVid	short-raw	0.089	0.091	**0.14**

Table 22. Results of the proposed systems on the MediaEval 2021 Predicting Media Memorability benchmark. We compare the results of a system variant without the Memorable Moments frame filtering method (R1), one with the Memorable Moments filtering included (R2), and the best performing systems from the MediaEval benchmarking task (MediaEval top).

Vision Transformer

We test two popular ViT architectures, namely the the DeiT [151] and the BEiT [152]. No special fusion method will be employed in this case, as these architectures will be tested in parallel and we will present the results for the best performing one. In the final stage, the features extracted via the Vision Transformer networks will be passed to a final regressor composed of a simple MLP head, similar to the one proposed in [150], that consists of three hidden layers of size 1024, 512 and 256 respectively.

Experimental results

We test the proposed methods on the new MediaEval 2021 Predicting Media Memorability benchmarking task [153]. This task presents participants with short and long-term memorability annotation tasks, measured over two different datasets, namely data extracted from the TRECVid dataset [154] and data extracted from the Memento10K dataset [155]. The dataset in this task will also provide us with the annotation necessary for separating the Memorable moments from the rest of the video.

The prediction subtask (subtask 1) measures system performance when training and testing on similar data (videos from the same dataset), while the generalization subtask (subtask 2) measures system perfomance when training and testing in different setups (i.e., training on Memento data and testing on TRECVid data). The results of the proposed system are presented in Table 22, where they are compared against the best results from this year's MediaEval campaign. The results show in most of the proposed runs, that the filtered approach represents an improvement over the non-filtered one, sometimes with a singnificant margin.

Relevant publications

- This method will be published in January-February 2022: `M.G. Constantin, B. Ionescu: "Using Vision Transformers and Memorable Moments for the Prediction of Video Memorability". Proceedings of MediaEval'21, February 2022.`

Relevant WP8 Use Cases

3C12 (Multimodal sentiment analysis). Memorability, as a measure of the ability of multimedia items of being stored and remembered by viewers is a concept that relates to the global user sentiment analysis.

8.3 DeepFusion Ensembling Systems

Late fusion, also known as ensembling systems or decision-level fusion, consists of a series of initial predictors, known as inducers, that are trained and tested on a certain dataset, and whose prediction outputs are fused or combined in a final phase to produce a improved set of predictions, that is a better representation of the ground truth values of the testing data than any of the individual inducers. These types of approaches have been found to be particularly beneficial in situations where single-system techniques do not perform well or there is an absolute need for near-perfect precision. While their utility has been demonstrated in some traditional tasks, such as video action recognition [145], there has recently been a noteworthy trend of using similar approaches in subjective tests that seek to understand how people perceive multimedia data, such as media memorability [156], violence detection [157] and media interestingness [158].

While there is a wide range of ensembling functions and methods for merging inducer prediction results, deep neural networks remain a novelty in this domain. Our work employing deep neural networks as the principal ensembling function is one of the first attempts in this manner, to the best of our knowledge. Several studies show that, thus far, DNN architectures have been generally used only as inducers, feature extractors or other intermediate modules in ensembling schemes, and never as the primary ensemble function [159]. Simple statistical techniques [160], such as late fusion via weighted arithmetic mean computation, voting systems, and so on, have dominated the process of ensembling thus far. Boosting algorithms like AdaBoost [161], Gradient Boosting [162] or XGBoost [163], Bagging [164] or Random Forests [165] are examples of implemented fusion approaches that require a learning step. While these methods have been successful in a variety of tasks, we theorize that by including deep neural networks as the primary ensembling function, late fusion results will greatly improve.

Problem definition

In a general sense, given a set of M dataset samples, and a series of N inducers, thus creating a vector $S = [s_1, s_2, ..., s_M]$ of samples and another vector $A = [a_1, a_2, ..., a_N]$ of inducers, a matrix Y can be created that represents the entire set of inducer predictions for the entire sample set as shown in Equation 9:

$$Y = \begin{bmatrix} y_{1,1} & \cdot & \cdot & \cdot & y_{1,N} \\ \cdot & \cdot & & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & \cdot \\ y_{M,1} & \cdot & \cdot & \cdot & y_{M,N} \end{bmatrix} \tag{9}$$

Furthermore, given a single random sample i, the set of predictions generated by the inducers can be defined as $[y_{i,1}, y_{i,2}, ..., y_{i,N}]$. The complexity and dimensionality of the individual $y_{i,j}$ values of course depends on the problem that is being solved: for one class regression these may represent single values, while for problems like ranking, multi-class regression or tagging, they may be represented by vectors of values.

We propose the following deep architectures for addressing the problem of deep ensembles: (i) dense architectures, (ii) attention augmented architectures, (iii) convolutional enhanced architectures, and (iv) Cross-Space-Fusion architectures.

Dense architectures

Dense architectures represent a straight-forward approach, where a variable number of dense (or fully connected) layers are linked together in order to create a simple yet effective structure that outputs the final result for the primary ensembling function. This architecture will also represent the final step for other types of architectures. While many different setups for this network can be implemented, we chose to go with a setup with variable network depth and width, and with inclusion or exclusion of batch normalization layers, as presented in Figure 32. Specifically, we vary the number layers in the network, testing values of $5, 10, 15, 20, 25$ and the number of neurons per layer, testing values of 25, 50, 500, 1000, 2000.

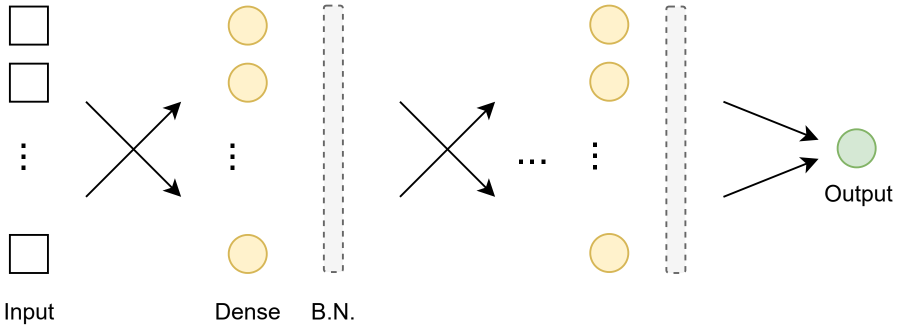


Figure 32. Deep fusion: Dense architectures, presenting the variable network width and depth, as well as the inclusion or exclusion of batch normalization layers.

Attention architectures

While in a general sense, attention mechanisms have the role of understanding and coding the parts of an image or general sample which are most important for the final prediction stage, in our particular case we use attention maps to encode and learn a set of weights that can be assigned to each individual inducer. For our particular application, we use a soft attention approach, that would create a vector of values $attn_i$ with values between 0 and 1, the system will create an appropriate attention mask $\widehat{attn_i}$, computed as the element wise product of the input vector and the attention vector:

$$\overline{y_i} = \begin{bmatrix} y_{i,1}, & y_{i,2}, & ..., & y_{i,N} \end{bmatrix} \tag{10}$$

$$\widehat{attn_i} = attn_i \odot \overline{y_i} \tag{11}$$

Convolutional architectures

Convolutional networks represented a big step forward for deep learning and, while the shape of the input space is not important, as one, two or three dimensional convolutional networks have been implemented, they still rely on spatial correlation between neighbouring elements from the input space. However the use of such layers would be interesting, especially for processing correlations

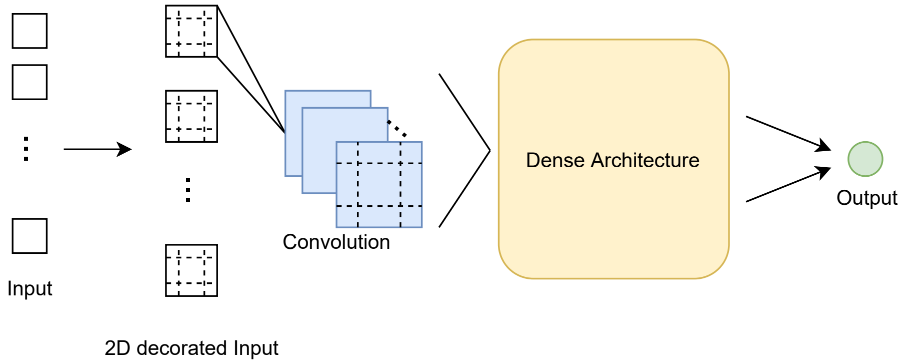


Figure 33. A schematic presentation of the Convolutional networks. The convolutional layer, is preceded by the input decoration stage and inserted into the dense architecture.

between individual inducers. Therefore, in order to utilize convolutional architectures, an input decoration method has to be developed, that can put correlated inducers in spatial proximity.

For starters, we consider the best function for defining the correlation between inducers to be the function that defines the metric of the task. Therefore, we will have the correlation between two individual inducers, m and n defined as $r_{m,n} = \mathcal{M}(p_m, p_n)$, where $\mathcal{M}$ represents the metric calculation function, and p_j represents the vector of outputs for the training set for inducer j. In the decoration step, each inducer output can be decorated with values that represent inducer outputs from the most correlated inducers, and with values that represent correlation scores between itself and the most correlated inducers. An input vector for a single sample i will therefore be transformed from a simple vector $[s_1, s_2, ..., s_N]$ to a more complex two-dimensional structure as follows:

$$\overline{dc_i} = \begin{bmatrix} r_{4,1} & c_{1,1} & r_{1,1} & . & . & . & r_{4,N} & c_{1,N} & r_{1,N} \\ c_{4,1} & s_1 & c_{2,1} & . & . & . & c_{4,N} & s_N & c_{2,N} \\ r_{3,1} & c_{3,1} & r_{2,1} & . & . & . & r_{3,N} & c_{3,N} & r_{2,N} \end{bmatrix} \tag{12}$$

where the pair $(c_{1,j}, r_{1,j})$ represents the inducer output and correlation score of the most similar inducer to c_j, $(c_{2,j}, r_{2,j})$ the second most similar, and so on.

Finally, 3 × 3 convolutional filters with a stride of 3 can be applied to this structure, followed by a pooling method, that could create a single score standing for each inducer, based not only on inducer output, but also on outputs from similar inducers and similarity scores. This type of structure is shown in Figure 33.

Cross-Space-Fusion architectures

While the convolutional approach can be successfully used to process correlation between inducers, we theorise that the nature of convolutions may not be appropriate for this type of data. Convolutional networks work under the assumption that the same convolutional filters can be applied to the entire input space in order to create better features that help understand spatial correlations in the data. However, in our particular case it may be necessary to learn completely different weights for each inducer centroid. Therefore, we propose a new decoration scheme, that would separate inducer outputs and inducer correlation scores in a third dimension, and that would create separate filters for each inducer centroid. Following the naming conventions presented in the convolutional approaches section, the following centroids will be built around each s_i element from the input space:

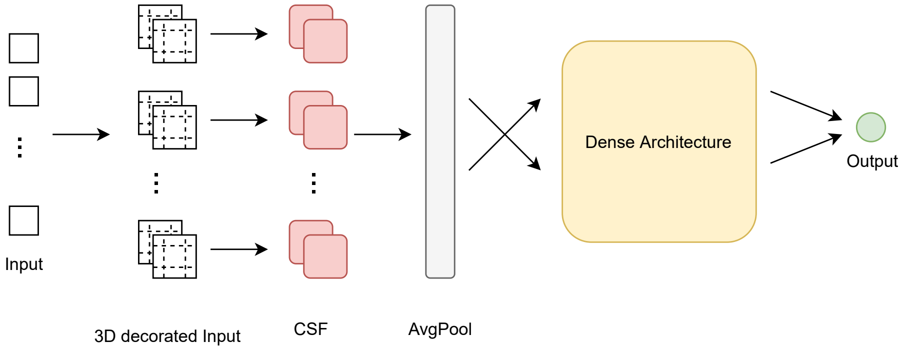


Figure 34. A schematic presentation of the Cross-Space-Fusion architecture. The Cross-Space-Fusion (CSF) layer is preceded by the input decoration scheme, and the results are passed to the Dense architecture after an average pooling operation.

$$C_i = \begin{bmatrix} c_{1,i} & c_{2,i} & c_{3,i} \\ c_{8,i} & s_i & c_{4,i} \\ c_{7,i} & c_{6,i} & c_{5,i} \end{bmatrix}, R_i = \begin{bmatrix} r_{1,i} & r_{2,i} & r_{3,i} \\ r_{8,i} & 1 & r_{4,i} \\ r_{7,i} & r_{6,i} & r_{5,i} \end{bmatrix} \tag{13}$$

Thus, the decorated input will now have a three-dimensional structure, and, for a single sample, the dimensions go from N to $(3N \times 3 \times 2)$. The novel Cross-Space-Fusion proposes to analyze these centroids by learning and optimizing two types of parameters for each centroid, namely α parameters, used for controlling inducer outputs and β parameters, used for controlling correlated inducers. These parameters will perform the following computations for each centroid:

$$\begin{bmatrix} \frac{\alpha_{1,i}\cdot s_i+\beta_{1,i}\cdot c_{1,i}\cdot r_{1,i}}{2} & \frac{\alpha_{2,i}\cdot s_i+\beta_{2,i}\cdot c_{2,i}\cdot r_{2,i}}{2} & \frac{\alpha_{3,i}\cdot s_i+\beta_{3,i}\cdot c_{3,i}\cdot r_{3,i}}{2} \\ \frac{\alpha_{8,i}\cdot s_i+\beta_{8,i}\cdot c_{8,i}\cdot r_{8,i}}{2} & s_i & \frac{\alpha_{4,i}\cdot s_i+\beta_{4,i}\cdot c_{4,i}\cdot r_{4,i}}{2} \\ \frac{\alpha_{7,i}\cdot s_i+\beta_{7,i}\cdot c_{7,i}\cdot r_{7,i}}{2} & \frac{\alpha_{6,i}\cdot s_i+\beta_{6,i}\cdot c_{6,i}\cdot r_{6,i}}{2} & \frac{\alpha_{5,i}\cdot s_i+\beta_{5,i}\cdot c_{5,i}\cdot r_{5,i}}{2} \end{bmatrix} \tag{14}$$

The Cross-Space-Fusion layer will therefore have 16 × N extra parameters to learn, where N represents the number of inducers, with 8 × N each of α and β parameters. An average pooling layer is inserted between the Cross-Space-Fusion part of the architecture and the dense layers, in order to produce a vector that can be classified by the dense architectures. This setup is presented in Figure 34.

Experimental results

We test these approaches on a large number of datasets and benchmarking tasks, in order to better understand their usefulness and generalization capabilities. We compare our results not only with the best performing inducers and best performing systems from the respective benchmarking tasks, but also with a set of traditional late fusion methods, namely statistical approaches [160], AdaBoost [161] and Gradient Boosting [162].

Furthermore, given the impossibility of creating a large number of inducers, training them and running them on a large number of datasets, we use the systems submitted by participants to the benchmarking tasks as the inducers for our systems. These systems were provided to us by the organizers of the various tasks and, considering that they only feature prediction values for the testing set, we have to create new splits in order to train our DeepFusion systems, and the traditional late fusion systems used as comparison baselines. We call these splits RSKF50, consisting of a random stratified k-fold split where 50% of the data is used for training and 50% for testing; and RSKF75, consisting of a random stratified k-fold split where 75% of the data is used for training and 25% for testing. In order to avoid the possibility of accidentally creating some ”lucky” splits, we run the splits several times, obtaining 100 total partitions. The final results for the fusion methods are presented as average results over the 100 splits.

The following datasets and benchmarking tasks are used for our experiments:

- The MediaEval 2015 Affective Impact of Movies [166] is a dataset and benchmarking task used for the detection of violent scenes in videos. The metric for this dataset is mean average precision (mAP), and 48 systems are used as inducers for violence prediction (denoted VSD2015).
- The MediaEval 2018 Emotional Impact of Movies [167] is a dataset and benchmarking task used for the prediction of the emotional content in videos, according to an arousal-valence dimension and to content that induces fear. The metric for the arousal-valence dimension is the mean squared error (MSE), while intersection-over-union (IoU) is used for fear prediction. A number of 30 systems are used as inducers for arousal and valence prediction (denoted Aro2018 and Val2018) and 18 systems for fear prediction (denoted Fear2018).
- Interestingness10k [142], previously presented in Section 8.1, representing methods collected during the 2017 edition of the MediaEval Predicting Media Interestingness task. Two tasks are used for this dataset, namely image and video-based prediction, both of them using mean average precision at 10 (mAP@10) as metric. A number of 33 systems are used as inducers for image prediction (denoted as INT10kImg) and 42 systems for video prediction (denoted INT10kVid).

	VSD2015	Aro2018	Val2018	Fear2018	INT10kImg	INT10kVid
	mAP	MSE	MSE	IoU	mAP@10	mAP@10
Benchmark	0.296	0.1334	0.0837	0.1575	0.1385	0.0827
SoA	0.303	0.1334	0.0837	0.1575	0.1985	0.093
Fusion	0.3521	0.1253	0.0783	0.1733	0.1523	0.0961
DF-Dense	0.6192	0.0571	0.0640	0.1938	0.2316	0.1563
DF-Attn	0.6228	**0.0568**	0.0640	0.1913	0.2399	0.1668
DF-Conv	**0.6281**	-	-	-	0.2293	**0.1692**
DF-CSF	-	**0.0568**	**0.0634**	**0.2091**	**0.2403**	0.1664

Table 23. Results for the DeepFusion ensembling networks compared with the best performers from the respective benchmarking competitions, state of the art on the datasets, and traditional fusion methods, under the RSKF50 setup.

The results for the RSKF50 and RSKF75 setups are presented in Tables 23 and 24 respectively, where they are being compared with the best results on the datasets recorded during the benchmarking competitions and in state-of-the-art literature, and against the chosen traditional late fusion methods. The DeepFusion architecture significantly surpasses the chosen baselines and the traditional methods, indicating the usefulness of such an approach, especially in cases where individual inducer results are not satisfactory. However, the current version of the DeepFusion network requires a high number of inducers in order to produce such significantly superior results, therefore heavily increasing the demand for hardware resources. Further studies must be made in order to find methods of reducing the need for a high number of inducers, or searching for methods of performing inducer selection and filtering.

	VSD2015	Aro2018	Val2018	Fear2018	INT10kImg	INT10kVid
	mAP	MSE	MSE	IoU	mAP@10	mAP@10
Benchmark	0.296	0.1334	0.0837	0.1575	0.1385	0.0827
SoA	0.303	0.1334	0.0837	0.1575	0.1985	0.093
Fusion	0.392	0.125	0.0783	0.1733	0.1674	0.1129
DF-Dense	0.6341	0.0549	0.0626	0.2129	0.3355	0.2677
DF-Attn	**0.6486**	0.0548	0.0626	0.2140	0.3389	0.2750
DF-Conv	0.6471	-	-	-	**0.3436**	0.2799
DF-CSF	-	**0.0543**	**0.0625**	**0.2242**	0.3408	**0.2825**

Table 24. Results for the DeepFusion ensembling networks compared with the best performers from the respective benchmarking competitions, state of the art on the datasets, and traditional fusion methods, under the RSKF75 setup.

Relevant publications

- M.G. Constantin, L.D. Ştefan, B. Ionescu : "Exploring Deep Fusion Ensembling for Automatic Visual Interestingness Prediction". In book Human Perception of Visual Information - Psychological and Computational Perspectives, Springer International Publishing, Eds. B. Ionescu, W. Bainbridge, N. Murray. December, 2021. [168].
  Zenodo record: https://zenodo.org/record/5006827#.YZzhkntBzTE.
- M.G. Constantin, L.D. Stefan, B. Ionescu : "DeepFusion: Deep Ensembles for Domain Independent System Fusion". International Conference on Multimedia Modeling - MMM 2021, June 22-24, Prague, Czech Republic, 2021. [169].
  Zenodo record: https://zenodo.org/record/5005938#.YZziF3tBzTE.

Relevant software and/or external resources

- Code for the DeepFusion ensembling method, containing all the types of architectures proposed as well as the input decoration methods can be found at https://github.com/cmihaigabriel/DeepFusionSystem_v2.

Relevant WP8 Use Cases

3C2-12 (Multimodal sentiment analysis). While we applied these methods and algorithms to several tasks related to the analysis of user perception and sentiment of multimedia (like interestingness, affect and violent content), we believe them to be general and would like to underline the possibility of applying them in other domains or use cases.

8.4 Pairwise Ranking Network for Affect Recognition

This method studies the problem of emotion recognition under the prism of preference learning. Affective datasets are typically annotated by assigning a single absolute label, i.e. a numerical value that describes the intensity of an emotional attribute, to each sample. Then, the majority of existing works on affect recognition employ sample-wise classification methods to predict affective states, using those annotations. We take a different approach and use a deep network architecture that performs joint training on the tasks of classification of samples and pairwise ranking between samples, inferring the ordinal relation of their corresponding affective labels. Our method is incorporated into existing affect recognition architectures and it is evaluated on datasets of electroencephalograms (EEG), leading to consistent performance gains.

Pairwise Ranking Network

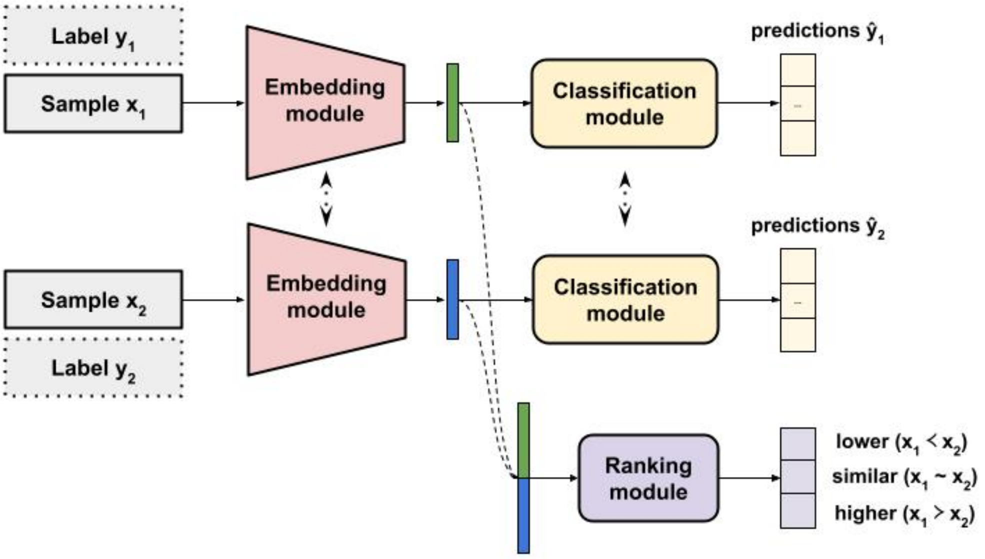


Figure 35. The architecture of a Pairwise Ranking Network that accomodates joint training on classification and ranking tasks.

The proposed methodology that derives pairwise ranking labels is applicable on datasets having as annotations either *continuous* affective ratings or *categorical* labels of ordinal nature. The ordinal relations for continuous ratings are shown in Table 25. Our method is simple and it can be integrated into existing affect recognition architectures. In essence, every deep neural network operating on the end-goal task of affect classification, consists of a backbone that extracts feature representations which are ultimately fed into a classification layer. We suggest adding an extra supervisory signal, by imposing a pairwise ranking objective on the intermediate representations learned by the backbone, leveraging the knowledge around the ordinal nature of emotions. The ranking task is performed by a ranking head that is stacked on top of the backbone network. The processing pipeline for classification remains intact and the total architecture is trained in an end-to-end manner. The classification and ranking losses are computed using a cross-entropy criterion.

Relation	Condition
$x_1 \succ x_2$	$y_1 > (y_2 + \epsilon)$
$x_1 \sim x_2$	$\|y_1 - y_2\| \leq \epsilon$
$x_1 \prec x_2$	$y_1 < (y_2 - \epsilon)$

Table 25. List of ordinal ranking relations and their corresponding conditions, when performing a comparison operation over continuous ratings.

Network architecture

Our architecture, named Pairwise Ranking Network ("PRNet"), can be seen in Fig. 35. The embedding module is the backbone of our architecture, serving as a feature extractor. The batch samples are fed as inputs to the embedding module and a feature embedding is computed for each sample. The produced embeddings are to be further processed for the tasks of classification and ranking, by the corresponding modules. The classification module receives as input the features produced by the embedding module, and predicts the affective state for each sample. The groundtruth targets are discrete emotion classes (e.g. "low"/"high" arousal, "low"/"high" valence). The ranking module operates on pairwise feature representations that correspond to sample pairs, and infers their ordinal relation with respect to their affective ratings. To form the pairwise feature representation of two samples, we get the feature vectors extracted from the embedding module for both samples, and we concatenate them across the channel dimension. To form multiple pairs of sample embeddings during training with a batch size of N_{b}, we split each batch into two sub-batches of size $N_{\mathrm{sub}} = \frac{N_{\mathrm{b}}}{2}$. Every sample of each sub-batch is compared against all samples of the other sub-batch, yielding $(N_{\mathrm{sub}})^2$ pairs in total. The total loss that is used to optimize the Pairwise Ranking Network is the sum of the classification and ranking losses.

Experimental results

We apply our method on two emotion recognition problems where the original affective annotations are inherently ordinal, aiming to exploit this property through our analysis. Specifically, we study the datasets of DEAP [170] and SEED [171].

Training details: Training is done for 20 epochs with a batch size of 40, using a Stochastic Gradient Descent (SGD) optimizer, learning rate $lr = 0.001$, momentum $m = 0.9$ and weight decay equal to 5e-4. For DEAP dataset, the ordinal ranking operation is performed setting $\epsilon = 0.25$. The training process is a subject-dependent 10-fold cross validation. The training process is subject-dependent, similarly to [171]. On both datasets, evaluation is done by computing the classification accuracy and F1 score. Our experiments explore the impact of joint training on the model classification performance. As a baseline method, a plain MLP network (with 2 FC layers in its embedding module and 1 FC classification layer) is trained only on the classification task. In our case, we train PRNet jointly on the classification and ranking tasks. From the results of Table 26 and Table 27, we can see that joint training improves the accuracy and F1 score both on the dataset of DEAP and SEED.

Model	Arousal		Valence	
	Acc.	F1	Acc.	F1
Classification loss	60.49	51.94	57.69	54.61
Proposed method: **Classification + ranking loss**	**60.60**	**53.25**	**58.42**	**55.57**

Table 26. Accuracy (%) and F1 score on DEAP dataset.

The results verify our motivation of forming and learning pairwise relations utilising the available affective annotations. On DEAP, we notice that collapsing fine-grained affective rating information into discrete classes, is harmful for the training process. Similarly, the fact that our approach considers the ordinality of the classes on SEED, shows that our method can be beneficial even in cases where the original annotations are discrete. The findings of our work highlight that exploring the ordinality of emotions through deep neural networks that accomodate pairwise ranking comparisons, is beneficial for affect recognition models. The proposed method is evaluated

Model	3-class problem	
	Acc.	**F1**
Classification loss	74.80	72.79
Proposed method: Classification + ranking loss	**76.98**	**75.51**

Table 27. Accuracy (%) and F1 score on SEED dataset.

on neurophysiological data with diverse affective annotation processes, showing consistent performance gains.

Relevant publications

- G. Zoumpourlis, I. Patras, Pairwise Ranking Network for Affect Recognition, 9th International Conference on Affective Computing and Intelligent Interaction (ACII 2021) [172] Zenodo record: https://zenodo.org/record/5550449.

Relevant WP8 Use Cases

3C2-13 (Modality-dependent sentiment analysis). Pairwise Ranking network is applicable to use-case 3C2-13, since it can be used to perform affect recognition.

8.5 Estimating continuous affect with label uncertainty

Continuous affect estimation is a problem where there is an inherent uncertainty and subjectivity in the labels that accompany data samples – typically, datasets use the average of multiple annotations or self-reporting to obtain ground truth labels. In this work, we propose a method for uncertainty-aware continuous affect estimation, that models explicitly the uncertainty of the ground truth label as a uni-variate Gaussian with mean equal to the ground truth label, and unknown variance. For each sample, the proposed neural network estimates not only the value of the target label (valence and arousal in our case), but also the variance. The network is trained with a loss that is defined as the KL-divergence between the estimation (valence/arousal) and the Gaussian around the ground truth. We show that, in two affect recognition problems with real data, the estimated variances are correlated with measures of uncertainty/error in the labels that are extracted by considering multiple annotations of the data.

Estimating with label uncertainty

When multiple annotations per sample are available (specifically in emotion and affect recognition), majority voting or averaging over the given multiple labels approaches are typically followed. Such methods, however, neglect the uncertainty that is inherent in such annotations introduced by multiple, usually disagreeing, annotators. Our method a) models the aforementioned uncertainty in the given annotations and b) uses it in order to predict both the (ground truth) mean value of the label and its (unknown) variance. An overview of the proposed method is shown in Fig. 36.

Methodology

Uncertainty aware regression: We begin by modelling the ground truth annotations as a set of independent uni-variate Gaussian distributions, for which we are given the true mean values

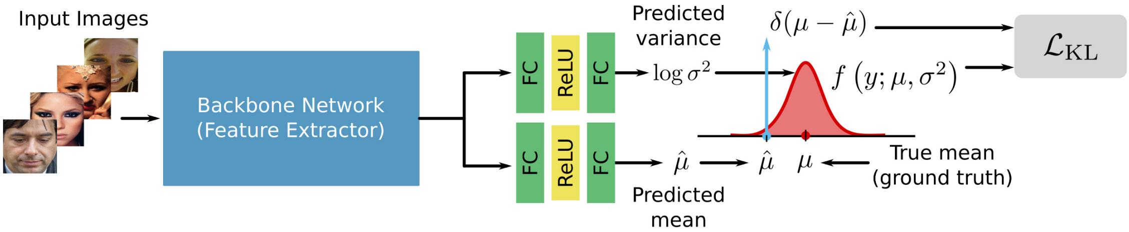


Figure 36. Proposed method overview: A backbone convolutional neural network is applied to input images in order to extract features which are subsequently used by two MLP heads in order to predict a) the variance σ^2 (top branch) and b) the mean $\hat{\mu}$ (bottom branch) of the annotation $y \sim \mathcal{N}\left(\mu, \sigma^2\right)$ for a given training sample. A KL-divergence loss function is then used to measure the difference between the Gaussian distribution $f(y; \mu, \sigma^2)$ and the Dirac delta distribution $\delta(\mu - \hat{\mu})$.

(ground truth), and we try to predict both the mean values and the corresponding variances. More specifically, let $y \sim \mathcal{N}\left(\mu, \sigma^2\right)$ denote an annotation label (e.g., the value of arousal for a given sample) with true mean value μ and unknown variance σ^2. For doing so, we jointly optimise a convolutional feature extractor backbone network and two MLP "heads", one predicting the mean and the other predicting the variance of the respective Gaussian, as shown in Fig. 36.

We achieve this by optimising a KL-divergence based loss function, $\mathcal{L}_{\text{KL}}$, which measures the difference between the predicted Gaussian, which is uniquely expressed by its true mean μ and the predicted variance σ^2 and its density is given by $f(y; \mu, \sigma^2)$, and a Dirac delta distribution centred at the predicted mean value $\hat{\mu}$, with density given by $\delta(\mu - \hat{\mu})$ (see Fig. 36).

In order to impose positivity on the predicted variance and avoid exploding gradients, we implicitly predict its Napierian logarithm, $s = \log \sigma^2$, and use it as $\exp(s) = \sigma^2$, as we will show below. That is, as shown in Fig. 36, the top MLP predicts the logarithm of σ^2.

By following similar arguments as in [173], we introduce a KL-divergence based loss function given by

$$\mathcal{L}_{\text{KL}} = \frac{(\mu - \hat{\mu})^2}{2\sigma^2} + \frac{\log \sigma^2}{2}, \tag{15}$$

when $|\mu - \hat{\mu}| \leq 1$, and by

$$\mathcal{L}_{\text{KL}} = \frac{1}{\sigma^2}\left(|\mu - \hat{\mu}| - \frac{1}{2}\right) + \log \sigma^2, \tag{16}$$

when $|\mu - \hat{\mu}| > 1$. That is, in the cases where the predicted mean values are far from their true values (typically during the early training process), we use the latter modified smooth $\mathcal{L}_1$ loss term shown in (16), while after achieving certain convergence we use the former fine-grained and uncertainty-aware loss term (15).

Architecture: In the case of continuous affect estimation on untrimmed videos, our basic architecture (Fig. 36) is set so as video features are obtained using a CNN with a trainable NetVLAD [174] layer. The NetVLAD architecture [174] is inspired by the Vector of Locally Aggregated Descriptors (VLAD), which is a pooling method that captures information about the statistics of local descriptors over the image, by storing the sum of residuals from cluster centers.

In this work, we modify the NetVLAD layer architecture to perform pooling along the temporal dimension, instead of the spatial. The input to the network is a set of pre-computed features, obtained during pre-training from each video frame. The network then performs a convolutional and average pooling operations followed by ReLU activation across the temporal dimension and then uses the NetVLAD layer as a pooling layer to standardise the feature vector size.

Experimental results

In order to assess the impact of the learned variances, we compare them with the corresponding variances induced by annotators disagreement – when multiple annotators' scores are available we can estimate uncertainty in the form of variance between annotators' scores. We propose to evaluate the learned variances against the annotator's variances at test time.

Table 28. PCC of learned variance and annotators variance on AMIGOS dataset

	Arousal	**Valence**
Proposed method	0.34	0.31

We compare the architecture against its baseline trained without variance prediction and an MSE loss. The architecture tested is simple uni-modal feed-forward networks as we aim to demonstrate the impact of uncertainty prediction.

Datasets: The AMIGOS dataset [175] consists of audio-visual and physiological responses of participants (either alone or in a group) to a video stimulus. In this work, we use the responses of individuals; 40 participants watched 16 short videos and 4 long ones. The former are defined as videos with length in the 50-150 second range. The responses are broken down to 20-second intervals and annotated by three annotators for *arousal* and *valence* on a scale from -1 to 1. We calculate the average score of the three annotators as the ground truth during training for the video segment. During testing, we use the variance of the annotators as an indication of uncertain or ambiguous samples and calculate the Pearson's Correlation Coefficient (PCC) between estimated and annotator's variance.

Metrics: The performance of the proposed methodology and the baselines is assessed using two evaluation metrics. For experiments conducted on the AMIGOS database [175], we report the Mean Square Error (MSE):

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(\mu_i - \hat{\mu}_i\right)^2, \tag{17}$$

where n is the number of videos in the database, μ_i is the ground truth and $\hat{\mu}_i$ is the predicted value. To better assess the performance of the regression task and to guarantee that results are comparable with other methods that apply transformations on the labels, we use Pearson's Correlation Coefficient (PCC), which for a pair of variables x, y with means $\bar{x}, \bar{y}$ is given by

$$\mathrm{PCC} = \frac{\sum_{i=1}^{n}\left(x_i - \bar{x}_i\right)\left(y_i - \bar{y}_i\right)}{\sqrt{\sum_{i=1}^{n}\left(x_i - \bar{x}_i\right)^2\left(y_i - \bar{y}_i\right)^2}}. \tag{18}$$

Implementation: The 1D convolutional and average pooling layers are set with a kernel size of 7 and stride 5 and the same number of channels according to the input. As we do not downsample frames in the video sequence, we assume neighbouring frames will have similar values and therefore implement a larger kernel and stride. The NetVLAD layer is initialised with 8 centroids. The training is performed in an end-to-end manner, and we follow a leave-one-subject-out cross validation protocol for each subject in the individual database, until the network converges. The network is trained using an ADAM optimiser with an initial learning rate of 0.01 multiplied by a factor of 0.1 every 100 epochs on two NVIDIA RTX 2080 GPUs.

Table 29. Results on AMIGOS using precomputed per frame Facial Action Units as input and a NetVLAD architecture.

	Arousal		**Valence**	
	MSE (std)	**PCC**	**MSE (std)**	**PCC**
NetVLAD	0.026 ($3e^{-}3$)	0.499	0.018 ($2e^{-}3$)	0.47
NetVLAD proposed	0.0354($6e^{-}3$)	**0.53**	0.018($2e^{-}3$)	**0.52**

Relevant publications

N. M. Foteinopoulou, C. Tzelepis, and I. Patras, 'Estimating continuous affect with uncertainty', presented at the 9th International Conference on Affective Computing & Intelligent Interaction (ACII), Nara, Japan, 2021.[176]
Zenodo record: https://zenodo.org/record/5714368

Relevant WP8 Use Cases

3C2-13 (Modality-dependent sentiment analysis). The presented method is applicable to use-case 3C2-13, since it can be used to perform affect recognition.

9 Real-life effects of private content sharing (T6.7)

Contributing partners: CEA, UPB

Personal data sharing via online social networks can lead to unexpected and, often times, serious consequences. These consequence are often difficult to predict at sharing time since the same information might be interpreted differently in various situations. For instance, a set of photos which depict someone partying is innocuous when shared with friends but might become detrimental if seen in a professional context. This task aims to design algorithms and associated tools which provide tangible feedback to users about the potential effects of data sharing. In the reported period, we focused on automatically assessing the effects of photo sharing in impactful real-life situations such as searching for an accommodation, a bank credit or a job. The proposed algorithm was integrated into a mobile app prototype which will be released in Android's Play Store.

The ubiquitous use of Online Social Networks (OSN) shows that their services are appealing to users. Most OSNs implement a business model in which access is free in exchange for user data monetization [177]. Intrusiveness is likely to grow with the wide usage of AI techniques to infer actionable information from users' data. Automatic inferences happen in the back-end of OSNs or of associated third parties and are not transparent for users. Data can be exploited in contexts unforeseen when sharing them initially. The main objective of our work is to improve user awareness about data processing through feedback contextualization. To do this, we introduce a plausible decision-making system which combines machine learning and domain knowledge.

User awareness is increased by linking the sharing process to impactful situations such as searching for a job, an accommodation, or a bank credit. Photos are in focus because they constitute a large part of shared data and contribute strongly to shaping user profiles [178]. The main technical contribution is a method that rates visual user profiles and individual photos in a given situation by exploiting situation models, visual detectors and a dedicated photographic profiles dataset. The proposed method, named *LERVUP* from **LE**arning to **R**ate **V**isual **U**ser **P**rofiles, learns a ranking of user profiles which attempts to reproduce human profiles ranking. *LERVUP* exploits a new descriptor which combines object impact ratings and object detection in a compact form. The contributions of objects with high ratings are boosted in order to mimic the way humans assess photographic content. We compare manual and automatic rankings of user profile ratings and obtain a positive correlation between them.

Crowdsourcing Object and Visual Profile Ratings

The interpretation of an object might vary between contexts, and so would the effects of sharing its images. Situations are modeled by crowdsourcing visual objects ratings. Impactful situations were selected: accommodation search (ACC below), bank credit demand (BANK), job search as IT engineer (IT) and job search as a waitress/waiter (WAIT). ACC and BANK are applicable to a large part of the population. IT and WAIT are relevant for population segments, but the respective job searches require different profiles. Detectable objects from the OpenImages [179], ImageNet [180] and COCO [181] datasets were rated to boost detector coverage. A limitation here is that task-relevant objects are missing and $\mathcal{D}$ could be enriched. A rating interface is created which includes for each situation: the object name, illustrative thumbnails and a 7-points Likert scale with ratings between -3 (strongly negative influence) to +3 (strongly positive influence). There were 56 participants in total, with 14 rating sets per situation. The final rating r is obtained by averaging their contributions. The resulting detection dataset $\mathcal{D}$ includes 269 objects with $r \neq 0$ for at least one situation. Inter-rater agreement, which is important for tasks prone to bias such as the one proposed here, is computed using the average deviation index (AD) [182]. The obtained

AD varies between 0.48 for IT and 0.65 for $WAIT$. These values are well below $AD \leq 1.2$, the maximum acceptable value for a 7-points Likert scale defined in [183]. The mean object ratings are -0.13 for BANK, 0.03 for ACC, 0.09 for IT and 0.27 for WAIT (standard deviations are 0.68, 0.7, 0.58 and 0.6 respectively). This illustrates the tendency of participants to be stricter when deciding about a bank loan than elsewhere. The is intuitive because granting a loan has tangible monetary consequences, which are easily internalized by participants. Inversely, WAIT, a situation with less serious implications, has the highest rating.

We collect manual ratings $m(U^i)$ for users U^i in situation $\mathcal{S}$ via crowdsourcing, again using a 7-points Likert scale. Ratings are collected from 9 participants for 500 users from the YFCC dataset [184] with 100 images per profile. YFCC was sampled because it includes images that were shared publicly under Creative Commons licenses which allow reuse. The images of each profile are shown on a single page, along with the possible situation rating. Participants were recruited via e-mail and details about the demographics of the panel are provided in the paper referred at the end of the section. Nine participants annotated each photographic profile. They were asked to look at all the photos and provide a global rating for each user in each situation. Inter-rater agreement is analyzed using the AD index [182]. AD values are 0.86 for ACC, 0.77 for BANK, 0.74 for IT and 0.83 for WAIT. These values are within the acceptability bounds defined in [183] ($AD \leq 1.2$).

Learning to Rate Visual User Profiles

We hypothesize that a supervised learning approach is better suited for profile rating. $LERVUP$ builds on a baseline which simply aggregates object ratings and detections and adds a descriptor compression followed by a training phase.

Image-level and user-level descriptors

Individual photos are a core factor in the manual rating of user profiles. It is thus interesting to aggregate object detection at the image level. Such a descriptor is equally interesting insofar that it provides understandable feedback about individual photo contributions to the profile rating. The descriptor includes three attributes which encode strong positive, strong negative and the average of the detection scores respectively.

Image-level descriptors are aggregated at user level to mimic the way in which humans rate visual user profiles. This is challenging because visual objects with different ratings appear in isolation or jointly in one or several profile images. Clustering is notably used in order to reduce the descriptor dimensionality. This is important in order to capture in a compact form patterns from an initial high-dimensional space defined by an array of object detectors and thus avoids the curse of dimensionality [185]. The proposed descriptor is an alternative to classical dimensionality reduction techniques [186], [187].

LERVUP training

Visual profile rating is modeled as a regression problem that exploits the user-level descriptor. $LERVUP$ training is deployed as a pipeline process. First, individual object detections are validated within each image. Second, the image-level descriptor is constructed per image. Third, clustering is applied to group together similar image descriptors and discover relevant patterns for the entire training set. Fourth, the discovered patterns are concatenated to build the user descriptor. Finally, a random forest regression model is used to learn the rating of visual user profiles. Random forest was chosen because it is robust to data that contain non-linear relationships between features and target variables [188]–[190].

	RCNN				MOBI			
	ACC	BANK	IT	WAIT	ACC	BANK	IT	WAIT
$BASE$	0.40	0.28	0.36	0.65	0.38	0.27	0.41	0.58
$BASE_{\eta}$	0.45	0.28	0.36	0.65	0.42	0.26	0.41	0.58
$BASE_{\eta}^{fr}$	0.45	0.33	0.36	0.65	0.42	0.30	0.41	0.58
$LERVUP$	0.48	0.48	0.46	0.66	0.44	0.27	0.47	**0.68**
$LERVUP^{fr}$	**0.55**	**0.50**	**0.50**	**0.68**	**0.49**	**0.42**	**0.51**	**0.68**

*Table 30. Pearson correlation between automatic and manual rankings of the ratings of visual user profiles. Results are presented with **RCNN** and **MOBI** as backbone networks for object detection. Best results in bold.*

Evaluation

The main objective of this first evaluation is to assess the feasibility of the task. Note that the user profiles dataset is not large enough to split it into train, validation and test subsets of sufficient sizes. We thus split the dataset in training and validation sets $\mathcal{L}$, and $\mathcal{V}$, which include 400 and 100 profiles, respectively. The optimal configuration of each method on $\mathcal{V}$ is obtained using grid search and reported below.

Object Detection Dataset and Models

The coverage ensured by the detection dataset is important to enable processing of different types of visual content. As we mentioned, we merge three existing datasets: OpenImages [179], ImageNet [180] and COCO [181]. Whenever an object is present in more than one dataset, a balanced sampling is performed. The resulting dataset includes 269 objects and 137,976 images. We limit imbalance by retaining at most 1,000 images per object. The average and standard deviation of the distribution are 513 and 305, respectively.

Detectors are trained with mobile and generic models. The mobile model (MOBI) is a MobileNetV2 [191] with depthwise convolutions, which offer a good precision/speed tradeoff. The detection head is a Single Shot MultiBox Detector [192], a fast single-stage method that is adapted for edge computation. The generic model (RCNN) uses Inception-ResNet-v2 [193] with atrous convolutions and a Faster RCNN module [194] for detection. While not designed specifically for mobile devices, tests showed that it is usable on recent Android smartphones.

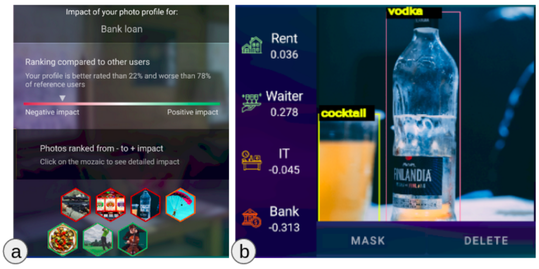
Methods

We test the following variants of the proposed methods:

- $BASE$ and $BASE_{\eta}$ - ranking based on a unique detection threshold and with threshold optimized per object and object selection.
- $BASE_{\eta}^{fr}$ - version of $BASE_{\eta}$, which exploits focal rating to boost salient objects.
- $LERVUP$ and $LERVUP^{fr}$ - proposed method without and focal rating activated, respectively.

Results

The performance of the different methods tested is presented in Table 30. All evaluated methods provide a positive correlation between manual and automatic rankings of the profile ratings, with a wide majority of reported correlations in the moderate (0.3-0.5 interval) or strong ranges (over



Impact of your photo profile for:

Bank loan

Ranking compared to other users

Your profile is better rated than 22% and worse than 78% of reference users

Photos ranked from - to + impact

Click on the mozaic to see detailed impact

Figure 37. Illustration of user feedback provided by the YDSYO app at profile level (a) and photo level (b).

0.5). This is a first positive result since the evaluated task is a complex one. The comparison of the two object detectors is globally favorable to RCNN. This result is intuitive insofar that RCNN is built with a higher capacity deep network architecture. The best global results are obtained with $LERVUP^{fr}$, which clearly outperforms the baselines. This finding validates the utility of the learning-based approach, which models automatic profile ranking as a regression problem. $LERVUP^{fr}$ is also better than $LERVUP$, with up 15 points gained over it (BANK with MOBI detector). The boosting of highly-rated objects via focal rating is thus validated.

The four modeled situations have variable performance. WAIT is the easiest situation (correlation up to 0.68) because the detection dataset contains a large number of food and beverage-related objects, which are often easy to detect. WAIT approximates an upper-bound performance one can expect with the available detection dataset. BANK is the most challenging situation tested, particularly for MOBI. More object detectors are required to improve results for this situation.

Conclusion

To summarize, we presented a new approach that unveils potential real-life effects of photo sharing. It is implemented for four situations but is extensible in terms of situations, types of data included, object detection models and profile rating methods. While promising, the approach is affected by a combination of human and technical biases. However, such biases are inherent to any AI-driven computer system and will also appear in real decision-making processes, which are mimicked here. The obtained results are promising insofar as the obtained correlations between human and automatic profile rankings are positive. The proposed method was integrated into YDSYO, a mobile app prototype [24], which will be released in the Android Play Store and is illustrated in Figure 37. The app provides feedback about the effects of photo sharing in each situation at profile and individual image levels. The comparison of the user profile rating to a set of reference profile ratings gives understandable feedback about where the user stands in the crowd in a particular situation. Individual photos are ranked by rating in that situation. The user can select each photo to have more details about its effects. Finally, a control mechanism is implemented by giving the option to mask or delete the photo.

Relevant publication

V. K. Nguyen, A. Popescu, and J. Deshayes-Chossart. Unveiling Real-Life Effects of Online Photo Sharing. WACV 2022.

[24] https://ydsyo.app

Relevant software and/or external resources

- Code: https://github.com/v18nguye/lervup_official
- Dataset: https://www.aicrowd.com/challenges/imageclef-2021-aware/

Relevant WP8 use case

After adaptation, the algorithms developed in this task could be used in use case 1E ”Capability for Trustworthy AI (by design)” which focuses on a grouping of audio-visual resources based on categories defined by each journalist.

10 Summary and Conclusion

This deliverable provides an overview of the research conducted in WP6: Human and Society-centred AI and presents the work of the individual partners in all seven subtasks. We showed that we covered a broad range of topics to make sure that AI4Media will be able to offer AI solutions to its use cases that are aware of possible societal challenges by assessing possible negative impact and actively countering it.

Regarding *Policy Recommendations for content moderation*, we put a spotlight on how automatic content moderation with its technological limitations fits into current and possibly future legislative frameworks. Further, our work emphasizes the urgent need of *Manipulation and synthetic content detection* and provides different approaches of detecting DeepFakes in text, audio and the visual domain. To show that alternative societal-friendly approaches are feasible in an integrated setting, we will develop a *Hybrid, privacy-enhanced recommendation* system for our use cases, that will integrate several approaches developed in AI4Media into a single system.

Four tasks are dedicated to the user's perception and the analysis of user generated content: Concerning *AI for Healthier political debate*, we showed how to automatically monitor public opinions and classify political tweets and discussions, also across languages. In the domain of *Perception of hyper-local news*, we turn to the analysis of health information in the context of Covid-19, building an own news corpus for later analysis and providing AI for local news analysis. *Measuring and Predicting User Perception of Social Media* is another crucial task that models how a user perceives certain content. Information that can be used later to assess how certain content tries to influence the end user or, on the other side of the spectrum, build systems that try to avoid this. Finally, we research the *Real-life-effects of private content sharing*, trying to automatically classify the impact a publicly shared private object may have on a users' privacy and developing a relevant app.

Most of the work here has already been published or submitted to relevant journals and conferences and we expect to have a positive impact of AI4Media in the research community. The next update of this deliverable (Second generation of Human- and Society-centered AI algorithms) will be published in month 36 of the project, in August 2023. The next deliverable produced in this work package will be exclusive on the results of T6.1: Policy Recommendations for content moderation in February 2023.

References

[1] A. Kuczerawy, "Safeguards for freedom of expression in the era of online gatekeeping," p. 19,

[2] J. Grimmelmann, "The Virtues of Moderation," LawArXiv, Preprint, May 2017. DOI: 10.31228/osf.io/qwxf5.

[3] K. Klonick, "The new governors: The people, rules, and processes governing online speech," *Harv. L. Rev.*, vol. 131, p. 1598, 2017.

[4] *Community standards enforcement — transparency center'*, https://transparency.fb.com/data/community-standards-enforcement/, [Accessed 06-Dec-2021].

[5] *Google Transparency Report — transparencyreport.google.com*, https://transparencyreport.google.com/youtube-policy/removals?hl=en, [Accessed 06-Dec-2021].

[6] *Rules Enforcement - Twitter Transparency Center — transparency.twitter.com*, https://transparency.twitter.com/en/reports/rules-enforcement.html, [Accessed 06-Dec-2021].

[7] E. Douek, "Governing Online Speech: From 'Posts-As-Trumps' to Proportionality and Probability," *SSRN Electronic Journal*, 2020, ISSN: 1556-5068. DOI: 10.2139/ssrn.3679607.

[8] T. Gillespie, *Custodians of the Internet.* Yale University Press, 2018.

[9] *Algorithmic content moderation: Technical and political challenges in the automation of platform governance - Robert Gorwa, Reuben Binns, Christian Katzenbach, 2020*, https://journals.sagepub.com/doi/full/10.1177/2053951719897945.

[10] E. Llansó, J. van Hoboken, P. Leerssen, and J. Harambam, "Artificial Intelligence, Content Moderation, and Freedom of Expression," p. 30,

[11] B. Heller, "Combating Terrorist-Related Content Through AI and Information Sharing," p. 8,

[12] M. Finck, "Issue paper: Artificial intelligence and online hate speech," 2019.

[13] *Photographer Nick Ut: The Napalm Girl — Buy Photos — AP Images — Collections*, http://www.apimages.com/Collection/Landing/Photographer-Nick-Ut-The-Napalm-Girl-/ebfc0a860aa946ba9e77eb786d46207e.

[14] "Fury over Facebook 'Napalm girl' censorship," *BBC News*, Sep. 2016.

[15] *Caught in the Net: The Impact of Extremist Speech Regulations on Human Rights Content*, https://syrianarchive.org/en/lost-found/impact-extremist-human-rights.

[16] C. B. Rishi Iyengar, *Facebook has language blind spots around the world that allow hate speech to flourish — cnn.com*, https://www.cnn.com/2021/10/26/tech/facebook-papers-language-hate-speech-international/index.html, [Accessed 06-Dec-2021].

[17] I. D. Raji, T. Gebru, M. Mitchell, J. Buolamwini, J. Lee, and E. Denton, "Saving Face: Investigating the Ethical Concerns of Facial Recognition Auditing," *arXiv:2001.00964 [cs]*, Jan. 2020. arXiv: 2001.00964 [cs].

[18] J. Buolamwini and T. Gebru, "Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification," p. 15,

[19] *OHCHR — Report of the Special Rapporteur to the General Assembly on AI and its impact on freedom of opinion and expression*, https://www.ohchr.org/EN/Issues/FreedomOpinion/Pages/ReportGA73.aspx.

[20] *Guidance Note on Content Moderation*, https://www.coe.int/en/web/freedom-expression/news/-/asset_publisher/thFVuWFiT2Lk/content/guidance-note-on-content-moderation.

[21] *Judgement of the court of 16 february 2022, c-360/10 sabam, par. 50*, https://curia.europa.eu/juris/document/document.jsf?text=&docid=119512&pageIndex=0&doclang=EN&mode=lst&dir=&occ=first&part=1&cid=323143, [Accessed 14-Dec-2021].

[22] *Joint letter on European Commission regulation on online terrorist content*, https://www.article19.org/resources/joint-letter-on-european-commission-regulation-on-online-terrorist-content/.

[23] T. Spoerri, "On Upload-Filters and other Competitive Advantages for Big Tech Companies under Article 17 of the Directive on Copyright in the Digital Single Market," *JIPITEC*, vol. 10, no. 2, Aug. 2019, ISSN: 2190-3387.

[24] r. D. S. et al Alex, "Online Platforms' Moderation of Illegal Content Online," p. 102,

[25] T. Quintel and C. Ullrich, "Self-Regulation of Fundamental Rights? The EU Code of Conduct on Hate Speech, Related Initiatives and Beyond," Social Science Research Network, Rochester, NY, SSRN Scholarly Paper ID 3298719, Oct. 2018.

[26] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and beyond: A survey of face manipulation and fake detection," *Information Fusion*, vol. 64, pp. 131–148, 2020.

[27] C. Stanciu and B. Ionescu, "Deepfake video detection with facial features and long-short term memory deep networks," in *ISSCS*, 2021.

[28] R. Tolosana, S. Romero-Tapiador, J. Fierrez, and R. Vera-Rodriguez, "Deepfakes evolution: Analysis of facial regions and fake detection performance," in *International Conference on Pattern Recognition*, Springer, 2021, pp. 442–456.

[29] Y. Li, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics," in *IEEE Conference on Computer Vision and Patten Recognition (CVPR)*, Seattle, WA, United States, 2020.

[30] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *ICCV 2019*, 2019.

[31] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Use of a capsule network to detect fake images and videos," *CoRR*, vol. abs/1910.12467, 2019. arXiv: 1910.12467. [Online]. Available: http://arxiv.org/abs/1910.12467.

[32] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, "Recurrent convolutional strategies for face manipulation detection in videos," in *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[33] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. K. Jain, "On the detection of digital face manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020.

[34] R. Tolosana, S. Romero-Tapiador, J. Fierrez, and R. Vera-Rodriguez, "Deepfakes evolution: Analysis of facial regions and fake detection performance," in *Pattern Recognition. ICPR International Workshops and Challenges*, Springer International Publishing, 2021, pp. 442–456.

[35] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo, "TV-L1 optical flow estimation," *Image Processing On Line*, vol. 2013, pp. 137–150, 2013.

[36] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[38] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[39] I. Amerini, L. Galteri, R. Caldelli, and A. Del Bimbo, "Deepfake video detection through optical flow based CNN," in *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct. 2019.

[40] R. Caldelli, L. Galteri, I. Amerini, and A. D. Bimbo, "Optical Flow based CNN for detection of unlearnt deepfake manipulations," Mar. 2021. DOI: 10.1016/j.patrec.2021.03.005. [Online]. Available: https://doi.org/10.1016/j.patrec.2021.03.005.

[41] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "Cnn-generated images are surprisingly easy to spot... for now," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8695–8704.

[42] F. Marra, C. Saltori, G. Boato, and L. Verdoliva, "Incremental learning for the detection and classification of gan-generated images," in *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, IEEE, 2019, pp. 1–6.

[43] R. Wang, F. Juefei-Xu, L. Ma, X. Xie, Y. Huang, J. Wang, and Y. Liu, "Fakespotter: A simple yet robust baseline for spotting ai-synthesized fake faces," in *International Joint Conferences on Artificial Intelligence Organization*, 2020.

[44] H. Guo, S. Hu, X. Wang, M.-C. Chang, and S. Lyu, "Eyes Tell All: Irregular Pupil Shapes Reveal GAN-generated Faces," *arXiv preprint arXiv:2109.00162*, 2021.

[45] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119.

[46] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," in *ICLR*, 2018.

[47] S. Gu, J. Bao, D. Chen, and F. Wen, "GIQA: Generated image quality assessment," in *European Conference on Computer Vision*, Springer, 2020, pp. 369–385.

[48] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, Dec. 2015.

[49] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *CVPR*, 2019.

[50] A. Gupta, *Human faces*, https://www.kaggle.com/ashwingupta3012/human-faces/version/1, 2020.

[51] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[52] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: A compact facial video forgery detection network," in *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, IEEE, 2018, pp. 1–7.

[53] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.

[54] M. Tan and Q. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” in *International Conference on Machine Learning*, PMLR, 2019, pp. 6105–6114.

[55] Q. Chen and V. Koltun, “Photographic image synthesis with cascaded refinement networks,” in *ICCV*, 2017.

[56] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *CVPR*, 2017.

[57] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, “High-resolution image synthesis and semantic manipulation with conditional gans,” in *CVPR*, 2018.

[58] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, “Semantic image synthesis with spatially-adaptive normalization,” in *CVPR*, 2019.

[59] X. Liu, G. Yin, J. Shao, X. Wang, *et al.*, “Learning to predict layout-to-image conditional convolutions for semantic image synthesis,” in *NeurIPS*, 2019.

[60] A. Dundar, K. Sapra, G. Liu, A. Tao, and B. Catanzaro, “Panoptic-based image synthesis,” in *CVPR*, 2020.

[61] L. Jiang, C. Zhang, M. Huang, C. Liu, J. Shi, and C. C. Loy, “Tsit: A simple and versatile framework for image-to-image translation,” in *ECCV*, 2020.

[62] H. Tang, D. Xu, Y. Yan, P. H. Torr, and N. Sebe, “Local class-specific and global image-level generative adversarial networks for semantic-guided scene generation,” in *CVPR*, 2020.

[63] H. Tang, S. Bai, and N. Sebe, “Dual attention gans for semantic image synthesis,” in *ACM MM*, 2020.

[64] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in *CVPR*, 2017.

[65] Q. Hou, L. Zhang, M.-M. Cheng, and J. Feng, “Strip pooling: Rethinking spatial pooling for scene parsing,” in *CVPR*, 2020.

[66] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, “The cityscapes dataset for semantic urban scene understanding,” in *CVPR*, 2016.

[67] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba, “Scene parsing through ade20k dataset,” in *CVPR*, 2017.

[68] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, “Deepfashion: Powering robust clothes recognition and retrieval with rich annotations,” in *CVPR*, 2016.

[69] C.-H. Lee, Z. Liu, L. Wu, and P. Luo, “Maskgan: Towards diverse and interactive facial image manipulation,” in *CVPR*, 2020.

[70] R. Tyleček and R. Šára, “Spatial pattern templates for recognition of objects with regular structure,” in *GCPR*, 2013.

[71] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” in *NeurIPS*, 2017.

[72] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *CVPR*, 2018.

[73] H. Tang and N. Sebe, “Layout-to-image translation with double pooling generative adversarial networks,” *IEEE Transactions on Image Processing*, vol. 30, pp. 7903–7913, 2021.

[74] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, “Fastspeech 2: Fast and high-quality end-to-end text to speech,” in *International Conference on Learning Representations (ICLR)*, Vienna, Austria, 2021.

[75] J. Kong, J. Kim, and J. Bae, “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis,” *arXiv preprint arXiv:2010.05646*, 2020.

[76] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. de Brébisson, Y. Bengio, and A. Courville, “Melgan: Generative adversarial networks for conditional waveform synthesis,” *arXiv preprint arXiv:1910.06711*, 2019.

[77] R. Reimao and V. Tzerpos, “For: A dataset for synthetic speech detection,” in *IEEE International Conference on Speech Technology and Human-Computer Dialogue (SpeD)*, Timisoara, Romania, 2019, pp. 1–10.

[78] J. Yamagishi, X. Wang, M. Todisco, M. Sahidullah, J. Patino, A. Nautsch, X. Liu, K. A. Lee, T. Kinnunen, N. Evans, and H. Delgado, “ASVspoof 2021: Accelerating progress in spoofed and deepfake speech detection,” in *ISCA Conference on ASVspoof 2021*, Virtual, 2021, pp. 47–54.

[79] W. Ping, K. Peng, K. Zhao, and Z. Song, “WaveFlow: A compact flow-based model for raw audio,” in *International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research (PMLR), Vienna, Austria, 2020, pp. 7706–7716.

[80] R. Prenger, R. Valle, and B. Catanzaro, “Waveglow: A flow-based generative network for speech synthesis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, 2019, pp. 3617–3621. DOI: 10.1109/ICASSP.2019.8683143.

[81] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, “Natural tts synthesis by conditioning wavenet on mel spectrogram predictions,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgari, Canada, 2018, pp. 4779–4783. DOI: 10.1109/ICASSP.2018.8461368.

[82] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, “Neural speech synthesis with transformer network,” in *AAAI*, Honolulu, HI, USA, 2019, pp. 6706–6713. DOI: 10.1609/aaai.v33i01.33016706.

[83] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2015. arXiv: 1409.1556 [cs.CV].

[84] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, M. Lucic, and A. Dosovitskiy, *Mlp-mixer: An all-mlp architecture for vision*, 2021. arXiv: 2105.01601 [cs.CV].

[85] S. Abdoli, P. Cardinal, and A. L. Koerich, “End-to-end environmental sound classification using a 1d convolutional neural network,” *Expert Systems with Applications*, vol. 136, pp. 252–263, 2019.

[86] M. Huzaifah, *Comparison of time-frequency representations for environmental sound classification using convolutional neural networks*, 2017. arXiv: 1706.07156 [cs.CV].

[87] C. Krätzer, A. Oermann, J. Dittmann, and A. Lang, “Digital audio forensics: A first practical evaluation on microphone and environment classification,” in *ACM Workshop on Multimedia & Security (MM&Sec)*, Dallas, TX, USA, 2007, pp. 63–74.

[88] L. Cuccovillo, S. Mann, M. Tagliasacchi, and P. Aichroth, “Audio tampering detection via microphone classification,” in *IEEE International Workshop on Multimedia Signal Processing (MMSP)*, Pula, Italy, Sep. 2013, pp. 177–182.

[89] L. Cuccovillo and P. Aichroth, “Open-set microphone classification via blind channel analysis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shangai, China, 2016, pp. 2074–2078.

[90] D. Luo, P. Korus, and J. Huang, “Band energy difference for source attribution in audio forensics,” *IEEE Transactions on Information Forensics and Security (TIFS)*, vol. 13, no. 9, pp. 2179–2189, 2018.

[91] M. A. Qamhan, H. Altaheri, A. H. Meftah, G. Muhammad, and Y. A. Alotaibi, “Digital audio forensics: Microphone and environment classification using deep learning,” *IEEE Access*, vol. 9, pp. 62 719–62 733, 2021.

[92] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising,” *IEEE Transactions on Image Processing (TIP)*, vol. 26, no. 7, pp. 3142–3155, 2017.

[93] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, “Librispeech: An asr corpus based on public domain audio books,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, QLD, Australia, 2015, pp. 5206–5210.

[94] C. Kotropoulos and S. Samaras, “Mobile phone identification using recorded speech signals,” in *IEEE International Conference on Digital Signal Processing*, Hong Kong, China, 2014, pp. 586–591.

[95] A. Giganti, “Speaker-independent microphone identification via blind channel estimation in noisy condition,” M.S. thesis, Politecnico di Milano, Milano, Italy, 2021.

[96] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, “Language models are few-shot learners,” in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[97] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, *Language models are unsupervised multitask learners*, 2019.

[98] N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, and R. Socher, *Ctrl: A conditional transformer language model for controllable generation*, 2019. arXiv: 1909.05858 [cs.CL].

[99] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, “Xlnet: Generalized autoregressive pretraining for language understanding,” in *Advances in Neural Information Processing Systems*, vol. 32, Curran Associates, Inc., 2019.

[100] T. Fagni, F. Falchi, M. Gambini, A. Martella, and M. Tesconi, “Tweepfake: About detecting deepfake tweets,” *PLOS ONE*, vol. 16, no. 5, pp. 1–16, May 2021. DOI: 10.1371/journal.pone.0251415.

[101] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, *Roberta: A robustly optimized bert pretraining approach*, 2019. arXiv: 1907.11692 [cs.CL].

[102] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems*, vol. 30, Curran Associates, Inc., 2017.

[103] S. Merity, N. S. Keskar, and R. Socher, “An Analysis of Neural Language Modeling at Multiple Scales,” *arXiv preprint arXiv:1803.08240*, 2018.

[104] ——, “Regularizing and Optimizing LSTM Language Models,” *arXiv preprint arXiv:1708.02182*, 2017.

[105] J. Bradbury, S. Merity, C. Xiong, and R. Socher, “Quasi-Recurrent Neural Networks,” *International Conference on Learning Representations (ICLR 2017)*, 2017.

[106] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” *arXiv preprint arXiv:1503.02531*, 2015.

[107] M. Phuong and C. Lampert, “Towards understanding knowledge distillation,” in *Proceedings of the International Conference on Machine Learning*, 2019.

[108] R. A. Potamias, G. Siolas, and A.-G. Stafylopatis, “A Transformer-based approach to irony and sarcasm detection,” *Neural Computing and Applications*, vol. 32, no. 23, pp. 17 309–17 320, 2020.

[109] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized BERT pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.

[110] S. Lai, L. Xu, K. Liu, and J. Zhao, “Recurrent Convolutional Neural Networks for text classification,” in *Proceedings of AAAI conference on artificial intelligence*, 2015.

[111] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” *arXiv preprint arXiv:1802.05365*, 2018.

[112] D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. John, N. Constant, M. Guajardo-Céspedes, S. Yuan, and C. Tar, “Universal sentence encoder,” *arXiv preprint arXiv:1803.11175*, 2018.

[113] S. I. Wang and C. D. Manning, “Baselines and bigrams: Simple, good sentiment and topic classification,” in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2012.

[114] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, “FastText.zip: Compressing text classification models,” *arXiv preprint arXiv:1612.03651*, 2016.

[115] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, “Xlnet: Generalized autoregressive pretraining for language understanding,” *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, 2019.

[116] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pretraining of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[117] F. Barbieri, F. Ronzano, and H. Saggion, “UPF-taln: SemEval 2015 Tasks 10 and 11. Sentiment analysis of literal and figurative language in Twitter,” in *Proceedings of the International Workshop on Semantic Evaluation (SemEval 2015)*, 2015.

[118] C. Ozdemir and S. Bergler, “CLaC-SentiPipe: Semeval2015 Subtasks 10 b, e, and Task 11,” in *Proceedings of the International Workshop on Semantic Evaluation (SemEval 2015)*, 2015.

[119] R.-A. Potamias, G. Siolas, and A. Stafylopatis, “A robust deep ensemble classifier for figurative language detection,” in *Proceedings of the International Conference on Engineering Applications of Neural Networks (EANN)*, Springer, 2019.

[120] R. Kiran, P. Kumar, and B. Bhasker, "OSLCFit (Organic Simultaneous LSTM and CNN Fit): A novel deep learning-based solution for sentiment polarity classification of reviews," *Expert Systems with Applications*, vol. 157, p. 113 488, 2020.

[121] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[122] J. Ling and R. Klinger, "An empirical, quantitative analysis of the differences between sarcasm and irony," in *Proceedings of the European Semantic Web Conference*, Springer, 2016.

[123] A. Ghosh, G. Li, T. Veale, P. Rosso, E. Shutova, J. Barnden, and A. Reyes, "Semeval-2015 Task 11: Sentiment analysis of figurative language in Twitter," in *Proceedings of the International Workshop on Semantic Evaluation (SemEval)*, 2015.

[124] H. Hewamalage, C. Bergmeir, and K. Bandara, "Recurrent neural networks for time series forecasting: Current status and future directions," *International Journal of Forecasting*, vol. 37, no. 1, pp. 388–427, 2021.

[125] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International journal of forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[126] D. Antonakaki, D. Spiliotopoulos, C. V. Samaras, P. Pratikakis, S. Ioannidis, and P. Fragopoulou, "Social media analysis during political turbulence," *PloS one*, vol. 12, no. 10, e0186836, 2017.

[127] A. Tsakalidis, S. Papadopoulos, R. Voskaki, K. Ioannidou, C. Boididou, A. I. Cristea, M. Liakata, and Y. Kompatsiaris, "Building and evaluating resources for sentiment analysis in the greek language," *Language resources and evaluation*, vol. 52, no. 4, pp. 1021–1044, 2018.

[128] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[129] J. Mahone, Q. Wang, P. Napoli, M. Weber, and K. McCollough, "Who's producing local journalism? assessing journalistic output across different outlet types," *DeWitt Wallace Center for Media and Democracy Report at Duke University*, 2019.

[130] *Wikipedia: List of newspapers in europe*, 2021. [Online]. Available: https://en.wikipedia.org/wiki/Lists_of_newspapers#Europe.

[131] J. Allen, B. Howland, M. Mobius, D. Rothschild, and D. J. Watts, "Evaluating the fake news problem at the scale of the information ecosystem," *Science Advances*, vol. 6, no. 14, eaay3539, 2020.

[132] *Covid-19 fake news dataset*, 2020. [Online]. Available: https://data.mendeley.com/datasets/zwfdmp5syg/1.

[133] *Covid19fn dataset*, 2020. [Online]. Available: https://data.mendeley.com/datasets/b96v5hmfv6/3.

[134] *Nlp for python*, 2020. [Online]. Available: https://raw.githubusercontent.com/susanli2016/NLP-with-Python/master/data/corona_fake.csv.

[135] *Fakecovid- a multilingual cross domain fact check dataset for covid-19*, 2020. [Online]. Available: https://zenodo.org/record/3965871.

[136] M. Del Río Carral, L. Volpato, C. Michoud, T.-T. Phan, and D. Gatica-Perez, "Professional youtubers' health videos as research material: Formulating a multi-method design in health psychology," *Methods in Psychology*, vol. 5, Dec. 2021.

[137] M. G. Constantin, M. Redi, G. Zen, and B. Ionescu, "Computational understanding of visual interestingness beyond semantics: Literature survey and analysis of covariates," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–37, 2019.

[138] A. Stevenson, *Oxford dictionary of English*. Oxford University Press, USA, 2010.

[139] D. E. Berlyne, "Interest as a psychological concept," *British Journal of Psychology*, vol. 39, no. 4, p. 184, 1949.

[140] P. J. Silvia, "What is interesting? exploring the appraisal structure of interest.," *Emotion*, vol. 5, no. 1, p. 89, 2005.

[141] ——, "Looking past pleasure: Anger, confusion, disgust, pride, surprise, and other unusual aesthetic emotions.," *Psychology of Aesthetics, Creativity, and the Arts*, vol. 3, no. 1, p. 48, 2009.

[142] M. G. Constantin, L.-D. Ştefan, B. Ionescu, N. Q. Duong, C.-H. Demarty, and M. Sjöberg, "Visual interestingness prediction: A benchmark framework and literature review," *International Journal of Computer Vision*, pp. 1–25, 2021.

[143] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[144] C. Liu, B. Zoph, M. Neumann, J. Shlens, W. Hua, L.-J. Li, L. Fei-Fei, A. Yuille, J. Huang, and K. Murphy, "Progressive neural architecture search," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 19–34.

[145] S. Sudhakaran, S. Escalera, and O. Lanz, "Gate-shift networks for video action recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

[146] D. Tran, H. Wang, L. Torresani, and M. Feiszli, "Video classification with channel-separated convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5552–5561.

[147] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 6450–6459.

[148] H. Touvron, A. Vedaldi, M. Douze, and H. Jégou, "Fixing the train-test resolution discrepancy," in *Advances in Neural Information Processing Systems*, 2019, pp. 8252–8262.

[149] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[150] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, *An image is worth 16x16 words: Transformers for image recognition at scale*, 2020. arXiv: 2010.11929 [cs.CV].

[151] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*, PMLR, 2021, pp. 10 347–10 357.

[152] H. Bao, L. Dong, and F. Wei, "Beit: Bert pre-training of image transformers," *arXiv preprint arXiv:2106.08254*, 2021.

[153] R. S. Kiziltepe, M. G. Constantin, C.-H. Demarty, G. Healy, C. Fosco, A. García Seco de Herrera, S. Halder, B. Ionescu, A. Matran-Fernandez, A. F. Smeaton, and L. Sweeney, "Overview of the mediaeval 2021 predicting media memorability task," in *Proceedings of MediaEval'21*, Bergen, Norway and Online, Dec. 2021.

[154] G. Awad, A. A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, A. Delgado, J. Zhang, E. Godard, L. Diduch, *et al.*, "Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval," *arXiv preprint arXiv:2009.09984*, 2020.

[155] A. Newman, C. Fosco, V. Casser, A. Lee, B. McNamara, and A. Oliva, "Multimodal memorability: Modeling effects of semantics and decay on video memorability," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, Springer, 2020, pp. 223–240.

[156] D. Azcona, E. Moreu, F. Hu, T. E. Ward, and A. F. Smeaton, "Predicting media memorability using ensemble models," in *Proceedings of the 2020 MediaEval Workshop*, CEUR Workshop Proceedings, 2020.

[157] Q. Dai, R.-W. Zhao, Z. Wu, X. Wang, Z. Gu, W. Wu, and Y.-G. Jiang, "Fudan-huawei at mediaeval 2015: Detecting violent scenes and affective impact in movies with deep learning.," in *Proceedings of the 2015 MediaEval Workshop*, 2015.

[158] S. Wang, S. Chen, J. Zhao, and Q. Jin, "Video interestingness prediction based on ranking model," in *Proceedings of the joint workshop of the 4th workshop on affective social multimedia computing and first multi-modal affective computing of large-scale multimedia data*, 2018, pp. 55–61.

[159] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, e1249, 2018.

[160] J. Kittler, M. Hatef, R. P. Duin, and J. Matas, "On combining classifiers," *IEEE transactions on pattern analysis and machine intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[161] Y. Freund, R. Schapire, and N. Abe, "A short introduction to boosting," *Journal-Japanese Society For Artificial Intelligence*, vol. 14, no. 771-780, p. 1612, 1999.

[162] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001.

[163] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[164] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[165] ——, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[166] M. Sjöberg, Y. Baveye, H. Wang, V. L. Quang, B. Ionescu, E. Dellandréa, M. Schedl, C.-H. Demarty, and L. Chen, "The mediaeval 2015 affective impact of movies task.," in *MediaEval*, 2015.

[167] E. Dellandréa, M. Huigsloot, L. Chen, Y. Baveye, Z. Xiao, and M. Sjöberg, "The mediaeval 2018 emotional impact of movies task.," in *MediaEval*, 2018.

[168] M. G. Constantin, L.-D. Ştefan, and B. Ionescu, "Exploring deep fusion ensembling for automatic visual interestingness prediction," *Human Perception of Visual Information: Psychological and Computational Perspectives. Springer*, 2021.

[169] ——, "Deepfusion: Deep ensembles for domain independent system fusion," in *International Conference on Multimedia Modeling*, Springer, 2021, pp. 240–252.

[170] S. Koelstra, C. Muhl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, and I. Patras, "Deap: A database for emotion analysis; using physiological signals," *IEEE transactions on affective computing*, vol. 3, no. 1, pp. 18–31, 2011.

[171] W.-L. Zheng and B.-L. Lu, "Investigating critical frequency bands and channels for eeg-based emotion recognition with deep neural networks," *IEEE Transactions on Autonomous Mental Development*, vol. 7, no. 3, pp. 162–175, 2015.

[172] G. Zoumpourlis and I. Patras, "Pairwise ranking network for affect recognition," in *9th International Conference on Affective Computing and Intelligent Interaction (ACII 2021)*, 2021.

[173] Y. He, C. Zhu, J. Wang, M. Savvides, and X. Zhang, "Bounding Box Regression With Uncertainty for Accurate Object Detection," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA: IEEE, Jun. 2019, pp. 2883–2892, ISBN: 978-1-72813-293-8. DOI: 10.1109/CVPR.2019.00300. [Online]. Available: https://ieeexplore.ieee.org/document/8953889/ (visited on 12/08/2020).

[174] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN Architecture for Weakly Supervised Place Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1437–1451, Jun. 2018, ISSN: 0162-8828, 2160-9292. DOI: 10.1109/TPAMI.2017.2711011. [Online]. Available: https://ieeexplore.ieee.org/document/7937898/ (visited on 07/21/2020).

[175] J. A. Miranda Correa, M. K. Abadi, N. Sebe, and I. Patras, "AMIGOS: A Dataset for Affect, Personality and Mood Research on Individuals and Groups," *IEEE Transactions on Affective Computing*, pp. 1–1, 2018, ISSN: 1949-3045, 2371-9850. DOI: 10.1109/TAFFC.2018.2884461. [Online]. Available: https://ieeexplore.ieee.org/document/8554112/ (visited on 11/11/2020).

[176] N. M. Foteinopoulou, C. Tzelepis, and I. Patras, "Estimating continuous affect with uncertainty," in *9th International Conference on Affective Computing and Intelligent Interaction (ACII 2021)*, 2021.

[177] K. Curran, S. Graham, and C. Temple, "Advertising on facebook," *International Journal of E-business development*, vol. 1, no. 1, pp. 26–33, 2011.

[178] S. Ahern, D. Eckles, N. Good, S. King, M. Naaman, and R. Nair, "Over-exposed?: Privacy patterns and considerations in online and mobile photo sharing," in *Proceedings of the 2007 Conference on Human Factors in Computing Systems, CHI 2007, San Jose, California, USA, April 28 - May 3, 2007*, M. B. Rosson and D. J. Gilmore, Eds., ACM, 2007, pp. 357–366. DOI: 10.1145/1240624.1240683. [Online]. Available: https://doi.org/10.1145/1240624.1240683.

[179] A. Kuznetsova, H. Rom, N. Alldrin, J. R. R. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig, and V. Ferrari, "The open images dataset V4: unified image classification, object detection, and visual relationship detection at scale," *CoRR*, vol. abs/1811.00982, 2018. arXiv: 1811.00982. [Online]. Available: http://arxiv.org/abs/1811.00982.

[180] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F.-F. Li, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015. DOI: 10.1007/s11263-015-0816-y. [Online]. Available: https://doi.org/10.1007/s11263-015-0816-y.

[181] T.-Y. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," in *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, D. J. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds., ser. Lecture Notes in Computer Science, vol. 8693, Springer, 2014, pp. 740–755. DOI: 10.1007/978-3-319-10602-1_48. [Online]. Available: https://doi.org/10.1007/978-3-319-10602-1%5C_48.

[182] M. J. Burke, L. M. Finkelstein, and M. S. Dusig, "On average deviation indices for estimating interrater agreement," *Organizational Research Methods*, vol. 2, no. 1, pp. 49–68, 1999.

[183] M. J. Burke and W. P. Dunlap, "Estimating interrater agreement with the average deviation index: A user's guide," *Organizational research methods*, vol. 5, no. 2, pp. 159–172, 2002.

[184] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "Yfcc100m: The new data in multimedia research," *Communications of the ACM*, vol. 59, no. 2, pp. 64–73, 2016.

[185] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE transactions on information theory*, vol. 14, no. 1, pp. 55–63, 1968.

[186] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *International conference on artificial neural networks*, Springer, 1997, pp. 583–588.

[187] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[188] M. Kayri, I. Kayri, and M. T. Gencoglu, "The performance comparison of multiple linear regression, random forest and artificial neural network by using photovoltaic and atmospheric data," in *2017 14th International Conference on Engineering of Modern Electric Systems (EMES)*, IEEE, 2017, pp. 1–4.

[189] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, "Random forest: A classification and regression tool for compound classification and qsar modeling," *Journal of chemical information and computer sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

[190] A. M. Youssef, H. R. Pourghasemi, Z. S. Pourtaghi, and M. M. Al-Katheeri, "Landslide susceptibility mapping using random forest, boosted regression tree, classification and regression tree, and general linear models and comparison of their performance at wadi tayyah basin, asir region, saudi arabia," *Landslides*, vol. 13, no. 5, pp. 839–856, 2016.

[191] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474.

[192] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C.-Y. Fu, and A. C. Berg, "SSD: single shot multibox detector," in *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., ser. Lecture Notes in Computer Science, vol. 9905, Springer, 2016, pp. 21–37. DOI: 10.1007/978-3-319-46448-0_2. [Online]. Available: https://doi.org/10.1007/978-3-319-46448-0%5C_2.

[193] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning,” in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, S. P. Singh and S. Markovitch, Eds., AAAI Press, 2017, pp. 4278–4284. [Online]. Available: http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14806.

[194] S. Ren, K. He, R. B. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., 2015, pp. 91–99. [Online]. Available: http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks.