Funded by the European Union
Horizon Europe
(HORIZON-CL4-2021-HUMAN-01-27
AI to fight disinformation)

Ref. Ares(2024)3260886 - 03/05/2024
www.ai4trust.eu

AI4TRUST

D3.1 -First release of AI tools for disinformation detection

PARTNERS

FONDAZIONE BRUNO KESSLER
CERTH CENTRE FOR RESEARCH & TECHNOLOGY HELLAS
UNIVERSITÀ DI TRENTO
NATIONAL CENTRE FOR SCIENTIFIC RESEARCH "DEMOKRITOS"
CNRS CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE
POLITEHNICA
GDI Global Disinformation Index
sky
SAHER EUROPE
DEMAGOG
MALDITA.ES
ASTIKI MI KERDOSKOPIKI ETAIRIA KENTRO KATAPOLEMISIS TIS PARAPLIROFORISIS / CIVIL NON-PROFIT COMPANY KENTRO KATAPOLEMISIS TIS PARAPLIROFORISIS
EURACTIV
ASOCIATIA DIGITAL BRIDGE
EUROPEJSKIE MEDIA SP ZOO
FINCONS GROUP
UNIVERSITY OF CAMBRIDGE

Project acronym	AI4TRUST
Project full title	AI-based-technologies for trustworthy solutions against disinformation
Grant info	ID 101070190-AI4TRUST
Funding	EU-funded under Digital, Industry, and space Overall budget € 5.950.682,50
Version	1.0
Status	Final Version
Dissemination level	Public
Due date of deliverable	30/04/2024
Actual submission date	03/05/2024
Work package	3
Lead partner for this deliverable	CERTH
Partner(s) contributing	CERTH, NCSR-D, UNITN, FBK, UPB, GDI
Main author(s)	Evlampios Apostolidis, Antonios Leventakis, Christos Koutlis, Symeon Papadopoulos, and Vasileios Mezaris (CERTH)
Contributor(s)	Georgios Petasis (NCSR-D), Christina Christodoulou, Sotiris Legkas, and Manthos Zidianakis (NSCR-D), Niculae Sebe, Elisa Ricci, Marco

	Formentini, and Raffaella Bernardi (UNITN), Sara Tonelli, Alan Ramponi, Margo Guerini, and Stefano Menini (FBK), Alexandru Caranica, Lucian Georgescu, Octavian Pascu, Adriana Stan, Dan Oneață, and Horia Cucu (UPB), Zoé Fourel (GDI)
Reviewer(s)	Riccardo Gallotti, Serena Bressan, and Danilo Giampiccolo (FBK), Marco Giovanelli (FINC)

Summary of modifications

VERSION	DATE	AUTHOR(S)	SUMMARY OF MAIN CHANGES
0.1	05/03/2024	Evlampios Apostolidis (CERTH)	Table of Contents and writing assignments
0.2	27/03/2024	All contributors	Added contributions according to the writing assignments
0.3	12/04/2024	Evlampios Apostolidis (CERTH)	Harmonized text and added comments for missing information
0.4	15/04/2024	Evlampios Apostolidis (CERTH)	Added “Introduction” and “Conclusions and next steps” parts
0.5	16/04/2024	All contributors	Added the requested information
0.6	20/04/2024	Marco Giovanelli (FINC), Riccardo Galloti (FBK)	Added comments after performing the Quality Assurance check on the document
0.7	25/04/2024	ALL	Revised version after addressing the comments from the QA, ready for final check before delivery to the PM team

0.8	26/04/2024	Evlampios Apostolidis (CERTH)	Final version delivered to the PM team for final check and submission to the European Commission
1.0	02-03/05/2024	Serena Bressan and Danilo Giampiccolo (FBK)	Review and final formatting of the deliverable before submission to the European Commission

Table of contents

List of abbreviations

ABBREVIATION	MEANING
ASR	Automatic Speech Recognition
CNN	Convolutional Neural Networks
DCT	Discrete Cosine Transform
DFDC	Deepfake Detection Challenge
DG	Domain Generalization
EER	Equal Error Rate
FPS	Farthest Point Sampling
LLM	Large Language Model
LSTM	Long-Short Term Memory
MFCC	Mel Frequency Cepstral Coefficients
MLM	Masked Language Modeling
MMD	Multimodal Misinformation Detection
NLP	Natural Language Processing
NSFW	Not Safe For Work
OLID	Offensive Language Identification Dataset
OOD	Out-Of-Distribution
PBE	Paint-By-Example
PFD	Prompt-Free-Diffusion
RC	Retrospection Consistency
RNN	Recurrent Neural Networks
TDNN	Time-Delay Neural Networks
SC	Style Consistency

SSL	Self Supervised Learning
STT	Speech-To-Text
SVM	Support Vector Machines
UI	User Interface
USMA	United States Military Academy
WER	Word Error Rate
XLMs	Cross-Lingual Language Models

Executive summary

Deliverable 3.1 - “First release of AI tools for disinformation detection” (hereinafter also referred to as D3.1) reports on the first release of the AI-driven data analysis methods developed in **WP3 - “AI-driven data analysis methods”** of the European project **AI4TRUST - “AI-based-technologies for trustworthy solutions against disinformation”**, and their integration into the AI4TRUST platform. It presents our solutions for text, audio, visual and multimodal analysis that will assist the detection of characteristics that are typically found in disinformation items, thus assisting the debunking of various types of fakes.

Moreover, it describes a set of generative methods that will be used to create data synthetically and support the training and evaluation of our methods for deepfake (image/video and audio) detection. Finally, it discusses the established plan for integrating a selected set of data analysis technologies into the Disinformation Warning System, which will be used to flag media items based on their likelihood to contain disinformation or not.

1. Introduction

Deliverable 3.1 - "First release of AI tools for disinformation detection" presents the first version of the developed technologies in **WP3 - "AI-driven data analysis methods"** since the beginning of the **AI4TRUST project**, and reports on the exposed APIs for the integration of some of these technologies in the AI4TRUST platform. Most of these technologies target the analysis of data from various modalities (e.g., text, audio, visual, multimodal) in order to spot characteristics that are typically found in disinformation items and assist the debunking of various types of fakes. In addition, a number of generative methods aim to create visual and audio data synthetically; these data will be used to further train and evaluate the developed deepfake (i.e., image/video and audio) detection technologies. Finally, selected data analysis technologies will be integrated in the Disinformation Warning System. This system will take into account the output of the integrated technologies and the GDI's data platform (i.e., index of verified and manipulated content), which is one of the partners in the AI4TRUST consortium, and provide an assessment stating whether a piece of content is likely to contain disinformation or not, with a confidence score.

The document is structured as follows: **Section 2** presents the developed technologies for detecting various disinformation signals in text (Section 2.1), spotting texts containing claims that are worthy of verification (Section 2.2), retrieving previously fact-checked claims that are similar to the investigated one (Section 2.3), and generating a verdict explaining why the claim under investigation can be considered true, only partially true or false (Section 2.4). Then, **Section 3** describes the provided set of models for multilingual speech-to-text transcription (Section 3.1), and the developed methods for deepfake audio detection (Section 3.2) and generation (Section 3.3). Following, **Section 4** reports on the released technologies for reverse video search on the Web (Section 4.1), deepfake image/video detection (Sections 4.2 and 4.3), sensational content detection (Section 4.4) and synthetic image/video generation (Section 4.5). Subsequently, **Section 5** presents the developed multimodal analysis methods for video anomaly detection (Section 5.1), audio anomaly detection (Section 5.2), visual-text misalignment detection (Section 5.3), and multimodal video deepfake detection (Section 5.4). Finally, **Section 6** discusses the conducted work on the development of the Disinformation Warning System, and **Section 7** concludes the document and reports on next steps.

2. Textual data analysis methods

2.1. Document Intelligence Level 1

2.1.1. Problem statement

In today's digital age, the spread of **disinformation** (hereinafter also referred to as “fake news” and/or “disinformation, misinformation, and maliformation”) through various communication platforms has become a significant problem. Disinformation can lead to the incitement of violence, the manipulation of public opinion, and harm to society, and affect both individuals and communities. To combat this issue, one of the AI4TRUST project’s initiatives is the **development of multilingual models** capable of detecting and classifying disinformation signals in textual content across eight languages. These models will differentiate between disinformation and non-disinformation, ensuring the input text is classified correctly.

Nevertheless, the detection of disinformation signals in text extends beyond the development of multilingual models. A significant challenge in this regard involves the **real-world application and testing of these models** to ascertain their effectiveness in accurately identifying and classifying disinformation signals across diverse scenarios. Another challenge pertains to the **availability and quality of datasets**, predominantly available in English and limited to specific disinformation signals. As a result, the lack of datasets in other languages, particularly low-resource languages, significantly hampers the development and fine-tuning of models that can effectively detect disinformation signals in these languages. It is, therefore, imperative to address these challenges to enable efficient detection of disinformation signals in text, irrespective of language and context.

The AI4TRUST project recognizes these challenges and is committed to a **comprehensive approach**. This includes effective moderation and intervention strategies, continuous testing, and refinement of models in real-world settings to ensure their effectiveness and reliability in supporting **fact-checkers, media professionals, and policymakers** to tackle disinformation.

2.1.2. Related work

The prevalence of disinformation in various forms, such as hate speech, offensive language, and clickbait, requires sustained efforts to prevent and address. Extensive research has been conducted in **machine learning and natural language processing (NLP)** to address these forms primarily in English. Previous studies have highlighted the need to differentiate between hate speech and offensive language, and divide them into subcategories to assist in the identification of insulting language. Davidson et al. (2017) introduced the hate speech detection dataset labeled as “hate speech", "offensive but not hate speech", and "neither offensive nor hate speech". They experimented with various traditional machine learning models, such as Logistic Regression, Naive Bayes, Decision Trees, Random Forests, and linear SVMs, and highlighted the difficulty in distinguishing between hate speech and offensive language. The 6th shared task of SemEval 2019 was introduced to identify offensive language on social media and was composed of three sub-tasks: detecting offensive language, identifying the type of offensive language, and determining the target of the offensive language. A variety of models were created for the task, including traditional machine learning methods, such as SVM and Logistic Regression, as well as deep learning models like CNN, RNN, BiLSTM, and BERT. These models were trained using the English Offensive Language Identification Dataset (OLID) (Zampieri et al., 2019). However, recent research has shown that the detection of offensive language and hate speech is no longer limited to English. Models have been developed to identify offensive language in Arabic, Danish, Greek and Turkish as part of Task 12 of SemEval 2020.

The most successful machine learning methods employed various versions of BERT, including BERT, RoBERTa, and mBERT (Zampieri et al., 2020). Furthermore, SemEval 2019's 5th task was focused on identifying hate speech towards immigrants and women on Twitter in both English and Spanish (Basile et al., 2019). The top-performing scores were achieved by SVMs, CNNs, and LSTMs. Research on clickbait detection is currently limited, with most studies focusing on English language datasets. For example, the SemEval 2023 Task 5 utilized the "Webis Clickbait Spoiling Corpus 2022" dataset (Hagen et al., 2022), which consisted of two sub-tasks: spoiler type classification and spoiler generation. The goal of these tasks was to determine the **most appropriate type of spoiler for a clickbait post** and to generate an actual spoiler for it (Fröbe et al., 2023). Most systems employed language models and performed few-shot learning or fine-tuning. However, research on clickbait detection in other languages is scarce, and there is a shortage of multilingual data available.

Due to **limited multilingual data**, extensive research has been conducted to enhance cross-lingual language understanding in such NLP tasks. The use of multilingual transformer-based masked language models such as mBERT (Devlin et al., 2019) and cross-lingual language models (XLMs) with Translation Language Modeling (TLM; Lample and Conneau, 2019) have demonstrated state-of-the-art results. Based on the work of Liu et al. (2019), which introduced RoBERTa, Conneau et al. (2020) created XLM-R, which significantly outperformed other multilingual models on several cross-lingual benchmarks. XLM-R is trained on 100 languages, with a large vocabulary of 250K sub-words, solely with Masked Language Modeling (MLM). The authors used the following fine-tuning techniques to evaluate the XLM-R's performance:

1. **Cross-lingual transfer:** They fine-tuned the multilingual model on English training set and evaluated on other languages.

2. **Translate-test:** They translated the development and test sets into English using NMT and fine-tuned a single English model.
3. **Translate-train (per-language):** They translated the English training set using NMT into each language and fine-tuned a multilingual model on each training set.
4. **Translate-train-all (multi-language):** They fine-tuned a multilingual model on the concatenation of all training sets from translate-train.

2.1.3. Proposed method

In AI4TRUST, we have developed multilingual models to detect disinformation signals in text through **binary text classification**. Initially, we focused on detecting specific disinformation signals such as hate speech (“HATE” or “NOT”), offensive language (“OFFENSIVE” or “NOT”), and clickbait as a type of sensational language (“CLICKBAIT” or “NOT”). These models support **eight European languages**, including English, Greek, Italian, Spanish, French, German, Polish, and Romanian.

Detecting hate speech and offensive language automatically is challenging due to the nuanced and context-dependent nature of these forms of communication. To address this issue, we developed separate **text classification models for hate speech detection and offensive language detection** to allow for more precise detection and categorization, as a text can be both hateful and offensive or only one of them. This precision is crucial for effective moderation and intervention strategies.

To develop these models, we collected and combined open-source annotated datasets for hate speech, offensive language, and clickbait in the English language. Influenced by the work from Conneau et al. (2020), we then translated these datasets into other languages using two Neural Machine Translation models, Opus-MT (Tiedemann et al. 2020) and M2M (Fan et al. 2020). Two, four and four datasets containing news and social media posts were combined for offensive language, hate speech and clickbait, respectively. Data pre-processing followed, which involved applying techniques such as data cleaning and removing duplicates, to ensure consistency and accuracy.

Throughout our research, we experimented with a **variety of multilingual models**, including Google’s mT5 (Xue et al., 2020) and flan-t5 (Won Chung et al., 2022), Microsoft’s XLM-align (Chi et al., 2021), infoXLM (Chi et al., 2021), mDeBERTa (He et al., 2023) and XLM-RoBERTa (Conneau, et al., 2020). We also used various cross-lingual training and validation techniques, including cross-lingual transfer and translate-train-all. The best approach was the translate- train-all technique. Moreover, we identified fine-tuning with all layers unfrozen as the best practice. The top-performing models, namely XLM-Roberta-Large and mDeBERTa-Base, were fine-tuned on a multilingual training set containing all the above-mentioned 8 languages. These models were evaluated on a multilingual validation and test set, as well as in each language separately.

2.1.4. Results and outlook

The performance of the **trained multilingual models** was assessed using various metrics, mainly Macro-F1 score. We computed an overall performance across all languages, as well as the performance for each language individually. The best models and their achieved performance are presented in Table 2.1.1 below. These models have been integrated into the exposed **API for detecting disinformation signals**, which can facilitate users in identifying hate speech, offensive language, and clickbait in texts. The models were evaluated based on **test sets**, but their efficacy will also be tested in **real-world scenarios** to assess their performance.

In the future, we plan to expand our research by developing multilingual models for argumentation mining and fact-checking. We will compare and assess the models' performance and deploy the best-performing ones.

Task	Model	Test Set Macro-F1 Score								
		All langs	En	El	De	It	Es	Ro	Fr	Pl
Hate Speech	mDeBERTa-base	88.16	88.61	87.40	87.92	88.25	87.34	87.59	87.43	90.90
Offensive Language	XLM-RoBERTa-Large	88.01	89.60	87.92	88.23	88.45	88.54	87.64	88.06	85.63
Clickbait	XLM-RoBERTa-Large	91.01	97.78	87.54	88.30	88.28	87.95	88.08	87.96	86.95

Table 2.1.1: Macro-F1 scores for best-performing disinformation signal detection models for text

2.1.5. Exposed API for integration

Our API leverages FastApi to offer **disinformation signal detection services** based on textual data to users. More specifically, the GET body returns a list of JSON objects that contain the available disinformation signals and their respective supported languages. The POST body accepts as input a SignalData JSON object whose content includes:

- The title (if there is one)
- The text
- The type of text (e.g article)
- The URL link

- The data source (e.g YouTube, Twitter)
- The language (en, el, it, es, de, fr, pl, ro)
- The selected disinformation signals

It returns a **list of DisinformationSignal JSON objects**. Each JSON object corresponds to a sentence. The text classification model predicts the selected disinformation signal/s and outputs a label for each sentence as well as a list with the following details concerning the predicted label:

- The text segment that was predicted
- The span of the text segment defined by start and end
- The textual output label of the classifier
- The confidence level of the predicted disinformation signal

More information is included in the API's detailed documentation. We have developed a **demonstration** using the Gradio platform that allows users to input text, choose a language, and select one or more disinformation signals among hate speech, offensive language, or clickbait, and then click the submit button. The model immediately identifies and highlights the chosen disinformation signal/s with a color. The API is ready for integration and it is available at: https://ai4trust.iit.demokritos.gr/ai4trust/docs#/**.** The Gradio demo is available at: https:// ai4trust.iit. demokritos.gr/demo/disinformation_signals/.

2.2. Check-worthy claim detection

2.2.1. Problem statement

Countering the spread of disinformation, misinformation, and malinformation is one of the major challenges of our society. However, human fact-checkers currently struggle to cope with the increasing amount of textual content being published. To facilitate human fact-checkers work in today's fast-paced information era, **automated tools that help professionals** focus on the subset of texts containing claims worthy of verification are of paramount importance.

Check-worthy claim detection is the first and a key task of the automated fact-checking pipeline (Guo et al., 2022). Specifically, it aims to detect texts presenting claims that are worthy of verification, i.e., those that appear to be false, may be of public interest or of impact to the public, or may cause harm to the society, entities, groups, or individuals (Nakov et al., 2022). Check-worthy texts contain claims that are both factual and verifiable, and therefore the task of detecting them indirectly acts as a filter for: i) the large number of non-factual and non-verifiable texts, i.e., those containing opinions only (non-fact-checkable), and ii) the claims that are not worthy of verification, i.e., those that can be easily checked by an average user (e.g., "Rome is the capital of Italy"), thus reducing the screening efforts of fact-checking professionals.

2.2.2. Related work

Check-worthy claim detection is an automated fact-checking subtask that has a long tradition in the NLP research field. A series of shared tasks and associated datasets have been proposed in recent years to raise awareness about the importance of advancing automated methods to support the fact-checking process. These shared tasks have been organized every year since 2018 in the context of the Conference and Labs of the Evaluation Forum (CLEF) initiative[1] under the lab name "CheckThat" (Barrón-Cedeño et al., 2023), each year with a set of subtasks covering different parts of the fact-checking pipeline. Notably, check-worthy claim detection is the only subtask that has been proposed at all six CheckThat editions (Barrón-Cedeño et al., 2023, Nakov et al., 2022, Nakov et al., 2021, Barrón-Cedeño et al., 2020, Elsayed et al., 2019, Nakov et al., 2018). Through those editions, datasets for training check-worthy claim detection models have been constantly extended to cover additional languages, starting from English and Arabic at CheckThat 2018 (Nakov et al., 2018) to Bulgarian, Turkish, Dutch, and Spanish at CheckThat 2022 (Nakov et al., 2022). The last edition instead makes an exception, covering English, Arabic, and Spanish only in multi-genre unimodal content (Barrón-Cedeño et al., 2023).

Regarding methods, state-of-the-art results for check-worthy claim detection are currently held by methods that rely on transformer-based (Vaswani et al., 2017) methods such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) and their language-specific variants, e.g., AraBERT (Antoun et al., 2020) for Arabic, typically using data manipulation or model ensembling strategies. The best results on the English portion of the CheckThat 2022 dataset are achieved by a RoBERTa model that leverages a back-translation-driven data augmentation process (Savchev et al., 2022), whereas best results for Spanish are achieved by fine-tuning a system based on mT5 (Xue et al., 2021) on multilingual data (Du & Gollapalli, 2022). Other proposed approaches in the field include the use of argumentative structure (Alhindi et al., 2021) and positive unlabeled learning (Wright & Augenstein, 2020).

Despite the notable progress in data and methods for check-worthy claim detection, there is currently **a lack of an extended coverage across languages and topics**. For instance, check-worthiness in Italian has never been studied due to the lack of annotated datasets. Our work in AI4TRUST fills this gap by creating the first annotated dataset for the Italian language and proposing a novel check-worthy claim detection model based on it. Moreover, current systems are based on techniques that by nature do not exploit the synergies between different but related tasks. Our work makes a step towards this goal by proposing **multi-task learning methods for check-worthy claim detection** that also leverage information about the factuality and verifiability of input texts.

[1] https://www.clef-initiative.eu/

2.2.3. Proposed method

We propose **multi-task learning models** that leverage information shared by different but related tasks to improve performance on check-worthy claim detection. Specifically, we **use pre-trained transformer-based models** as our encoders, on top of which we devise two decoders: one dedicated to factuality/verifiability and one for check-worthiness. Intuitively, since check-worthiness can be assessed only if an input text is factual and verifiable, having both tasks interplaying should lead the model to exploit the inductive bias of the factuality/verifiability task to improve decisions on the check-worthiness task. Decoders use softmax with a cross entropy loss and we output both the predicted class and its score.

Datasets. For Italian, there is currently no dataset for check-worthy claim detection in literature. Motivated by the lack of resources for Italian, we created a novel annotated corpus specifically for AI4TRUST. The dataset covers "migration", "climate change" and "public health" topics across a six-year period to minimize topic and temporal biases and has been annotated for both factuality/verifiability and check-worthiness by two annotators with diverse backgrounds and socio-demographic characteristics to embrace different perspectives. It consists of 2,160 annotated posts from Twitter, balanced across topics and time periods (720 per topic, 360 per year). The gold labels have been determined as follows: i) for factuality/verifiability, we consider the instance as factual/verifiable if both annotators agreed on it; and ii) for check-worthiness, we consider the post check-worthy if both annotators have labeled the instance with at least "probably yes" among the following ordered options: "definitely no", "probably no", "ambivalent", "probably yes", and "definitely yes". For English, we instead use subtask 1A and 1B data from the CheckThat 2022 lab (Nakov et al., 2022), namely the datasets for "check-worthiness estimation" and "verifiable factual claims detection" tasks and merge them for exploiting the inter-relations between the tasks in the modeling phase.

Pre-processing. Data minimization has been applied both to Italian and English posts by replacing possible user mentions, email addresses, URLs and phone numbers in the post text with placeholders (i.e., [USER], [EMAIL], [URL], and [PHONE], respectively). We also lowercase the texts to mitigate data sparsity.

Experimental setup. For Italian, we rely on the language-specific AlBERTo (Polignano et al., 2019) and UmBERTo (Parisi et al., 2020) language models as our encoders. For English, we instead use BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019). We employ MaChAmp (van der Goot et al., 2021), a toolkit for multi-task learning in NLP, and experiment with single-task and multi-task learning setups with default hyper-parameters as detailed in van der Goot et al. (2021). For multi-task learning experiments, we also assess different loss weights for the auxiliary task (i.e., factuality/verifiability), namely 0.1, 0.5, and 1.0. As regards the data splits, for Italian we use stratified *k*-fold cross validation (*k*=5) and report average scores and their standard deviation, whereas for English we use the standard splits provided by the CheckThat lab organizers (Nakov et al., 2022).

2.2.4. Results and outlook

Check-worthy claim detection is typically framed as a **binary classification task** (i.e., the texts being labeled as check-worthy *vs* non check-worthy) or as a **ranking task** (i.e., the texts being sorted by a check-worthiness score for prioritization), and systems' performance is evaluated through F1 score for the positive (i.e., check-worthy) class (pos F1) or mean average precision (MAP), respectively (Panchendrarajan & Zubiaga, 2024).[2] In previous CheckThat competitions, only one or the other have been typically employed. For the sake of comprehensive evaluation, we report both of them for all models and their variants.

As we can see in Table 2.2.1, for Italian, all multi-task learning variants outperform single-task models across both metrics. The best encoder for the task is UmBERTo, whereas the loss weight for the factuality/verifiability task that leads to the highest performance is 0.1. This means that factuality/verifiability information is useful for check-worthiness if treated as an auxiliary task. Similar findings apply for English, but the best encoder for check-worthy claim detection is RoBERTa – a transformer-based model pre-trained training on English texts – which achieved a 0.6754 MAP score and 0.6355 F1 score for the positive class.

As next steps, we plan to extend check-worthy claim detection to the Spanish language using publicly available data. Moreover, we aim to experiment with the aforementioned models using balanced class weights to better deal with label imbalance and perform additional tuning.

Setup	Model	MTL parameters	Pos F1	MAP
Random baseline			0.3986±$_{0.03}$	0.3053±$_{0.02}$
Single task	AlBERTo	–	0.6850±$_{0.05}$	0.7544±$_{0.04}$
Multi-task	AlBERTo	Aux loss weight: 0.1	0.6875±$_{0.05}$	0.7593±$_{0.04}$
Multi-task	AlBERTo	Aux loss weight: 0.5	0.7031±$_{0.03}$	0.7677±$_{0.03}$
Multi-task	AlBERTo	Aux loss weight: 1.0	0.7163±$_{0.02}$	0.7741±$_{0.02}$
Single task	UmBERTo	–	0.7121±$_{0.03}$	0.7937±$_{0.04}$
Multi-task	UmBERTo	Aux loss weight: 0.1	**0.7242**±$_{0.03}$	**0.8004**±$_{0.04}$
Multi-task	UmBERTo	Aux loss weight: 0.5	0.7240±$_{0.02}$	0.7957±$_{0.03}$
Multi-task	UmBERTo	Aux loss weight: 1.0	0.7216±$_{0.03}$	0.7947±$_{0.04}$

[2] Indeed, given that check-worthy claims are typically a minority, leading to a high-class imbalance in the dataset, metrics such as micro F1 score or accuracy are not suitable for assessing systems' performance on the task.

Table 2.2.1: Results for check-worthy claim detection in Italian across setups, models, and MTL (multi-task learning) parameters. We report average scores across the *k*=5 splits and standard deviation. The three best-performing configurations are underlined, whereas the best one is both underlined and in bold.

2.2.5. Exposed API for integration

We expose an **API for check-worthy claim detection** in both Italian and English. Given an input text, we return both a check-worthiness label and its associated score. The check-worthiness label is 1 if the text is check-worthy and 0 if it is not, and the check-worthiness score represents the score for that label (in [0, 1]).[3] The score allows us to better assist the fact-checkers in prioritizing the most relevant texts (i.e., those with label=1 and the highest check-worthiness score). In the following, we describe the required entry point of the implementation (e.g., required function parameters), request parameters and outputs, along with an example. The technical documentation is made available at: https://dh.fbk.eu/ai4trust-api/docs#/.

Entry point

```
https://dh.fbk.eu/ai4trust-api/check-worthiness/vX.Y/
```

where vX.Y is the version of the API (i.e., v2.0 as of 2024-03-28).

To access the APIs, a bearer token must be requested and provided in the request header.

Request parameters

- **text**: the input text
- **lang**: the language of the input text (i.e., either `it` or `en`)

An example of the input (in JSON format) is presented in the following:

```
{
    "text": "In 10 anni tagliati più di 500 ospedali, 155.000 posti letto,
    10 mila medici e 31 mila infermieri. In 10 anni aperti 29.485 centri di
    accoglienza per immigrati. Noi veniamo sempre per ultimi. Non c'è
    Nient'altro                   da                  aggiungere.",
    "lang":                                                   "it"
}
```

Output

- **label**: 1 if check-worthy, 0 if not check-worthy

[3] A text with label=1 and score=0.86 means that the text is check-worthy at 0.86 and not check-worthy at 1-0.86=0.14, whereas a text with label=0 and check-worthiness score=0.92 means that the post is check-worthy at 1-0.92=0.08 and not check-worthy at 0.92).

- **`score`**: a number ranging from 0 to 1 denoting the score for the predicted label

An example of the output (in JSON format) is presented in the following:

```
{
    "label": "1",
    "score": "0.8628964424133301"
}
```

2.3. Fact-checked claim retrieval

2.3.1. Problem statement

Among the tasks carried out by fact-checkers, **assessing whether a claim has already been fact-checked in the past to avoid replicating previous efforts** is particularly relevant, since it would enable them to use available (limited) resources more efficiently. This is even more important if we consider the possibility to exploit in a unified solution information about previously fact-checked claims, even in multiple languages, creating a bridge across fact-checking agencies in different countries.

The task of retrieving previously fact-checked claims has been therefore proposed to address this need, using past information about already fact-checked claims as a knowledge store against which new claims can be compared. While performing the task in a monolingual setting, where past claim information and new claims are both in the same language, may be quite straightforward using current NLP transformer-based techniques, foreseeing a multilingual scenario, with input claims and database of fact-checks in different languages, is more challenging and requires **innovative solutions across information-retrieval** and **approaches supporting cross-lingual similarity**.

2.3.2. Related work

Despite its relevance for fact-checking, multilingual fact-checked claim retrieval is still an **under-explored area or research**. In fact, previous work mostly focuses on post-claim pairs in English only (Shaar et al., 2020, Hardalov et al., 2022) or in a small set of languages from specific geographical areas (e.g., languages of India such as Bengal, Hindi, Malayalam, and Tamil in Kazemi et al. (2021)). Very recently, Pikuliak et al., (2023) took a step forward and released MultiClaim, a multilingual dataset consisting of over 205k fact-check titles and 28k social media posts with 31k post–fact-check pairs. Posts have been retrieved and associated with a fact-check if the very fact-checking article explicitly reviewed a post. Although many languages in MultiClaim are under-represented (i.e., <50 entries) or are not spoken in Europe, we rely on this dataset in our experiments since it is the most complete fact-checked claim retrieval dataset to date and provides a valuable resource for developing methods for the main European languages.

Regarding methods, BM25 (Robertson & Zaragoza 2009) still represents a strong baseline in fact-checked claim retrieval. Recent successful methods include a **variety of neural text embeddings models** based on Sentence-BERT (Reimers & Gurevych, 2019). As expected, the performance of these pre-trained models has been shown to increase with additional fine-tuning on data for the task at hand (Pikuliak et al., 2023), but the margin is still very limited in the case of previously fact-checked claim retrieval. Moreover, the selection of negative examples for robust fine-tuning is still an open area of research, and in our work, we aim to fully leverage findings on those aspects to **improve the performance of multilingual models** on the task.

2.3.3. Proposed method

We are currently working on **a multilingual method based on recent sentence embedding models for the retrieval of already fact-checked claims**, whose first version with associated API is planned to be released by August 31, 2024 (D6.2). We have already managed to reproduce the baseline approaches from Pikuliak et al. (2023), namely unsupervised and supervised training of recent Sentence-BERT models (e.g., GTR-T5; Ni et al., 2022) and MiniLM-L12 (Wang et al., 2020), and we are currently investigating novel methods for meaningfully selecting negative examples for fine-tuning. Specifically, our goal is to automatically select challenging negative examples for fine-tuning, in order to avoid saturating the training set (and thus, performance) after just a few training iterations. We are currently experimenting with approaches based on topical and metadata features. Below we summarize the data used and the preprocessing stages:

Dataset. We use MultiClaim (Pikuliak et al., 2023) for our experiments since it represents the more comprehensive dataset for multilingual fact-checked claim retrieval to date.

Pre-processing. We converted the original dataset to a more easily processable .tsv format, with a column dedicated to each unit of information. Moreover, we applied data minimisation to both claims and social media posts by replacing possible user mentions, email addresses, URLs, and phone numbers with placeholders (i.e., [USER], [EMAIL], [URL], and [PHONE], respectively). We also lowercase the texts to mitigate data sparsity.

2.3.4. Results and outlook

The tool is still in a development phase, but we have already managed to reproduce the results from the Pikuliak et al. (2023) paper as our baselines. Our results on English using GTR-T5 (Ni et al., 2022) are reported in Table 2.3.1. We note that our experimental setup for the release of the tool is different from the original one introduced in Pikuliak et al. (2023). Specifically, we perform

data minimization before training, and this may lead to results that slightly differ from the ones reported in their work. We believe that data minimization is an important step not only for robustness reasons, i.e., avoiding models to exploit shortcuts / spurious associations (e.g., usernames) for making a prediction, but also for preserving users' anonymity.

Future developments on the tool include the: i) **experimentation with additional languages** (namely Spanish, German, and French), and ii) **design of novel negative sampling strategies** to improve performance in the supervised scenario.

Setup	Pair S@10	Post S@10	MAP@5
Unsupervised	0.7018±$_{0.02}$	0.7124±$_{0.02}$	0.6827±$_{0.02}$
Supervised	**0.7132±$_{0.01}$**	**0.7244±$_{0.01}$**	**0.6879±$_{0.02}$**

Table 2.3.1: Results for fact-checked claim retrieval in English across setups using GTR-T5 baselines. We report average scores across the *k*=5 splits and standard deviation. The best-performing setup is in bold.

2.3.5. Exposed API for integration

Even if models for fact-checked claim retrieval are currently in a development phase, we have already started **designing the API for fact-checked claim retrieval.** Given an input text, we will return the most similar, already-verified claims to the ones expressed in the input. The API will provide a similarity score for input–claim pairs, and the AI4TRUST platform will be in charge to show up to *N*=3 most similar claims (in a decreasing order). We describe below the required entry point of the implementation (e.g., required function parameters), request parameters and outputs, along with an example. The technical documentation will be made available once the first version of the tool will be released.

Entry point

```
https://dh.fbk.eu/ai4trust-api/claim-retrieval/vX.Y/
```

where vX.Y is the version of the API (i.e., it will be v1.0 at the first release on 2024-08-31).

To access the APIs, a bearer token must be requested and provided in the request header.

Request parameters

- `text`: the input text
- `lang`: the language of the input text (ISO 639-2 code)

An example of the input (in JSON format) is presented in the following:

```
{
    text: "Of course the climate is changing. It always has. It always will.",
    lang: "en"
}
```

Output

- **`claims`**: a list containing the texts of the most similar previously fact-checked claims
- **`scores`**: a list containing the similarity scores for the texts in `claims`

An example of the output (in JSON format) is presented in the following:

```
{
    "claims": [
        "There has always been climate change",
        "Climate change is not happening."
    ],
    "scores": [
        "0.9664827774927486",
        "0.9458690288845028"
    ]
}
```

2.4. Verdict generation

2.4.1. Problem statement

When fighting the proliferation of fake news, being able to **assess the veracity of the numerous disinformation claims spreading online** is a very important, yet time-consuming task. According to the above-mentioned claim, fact-checkers can be required to read through a large amount of sources in order to confirm or confute the claim. In a context where both the number of claims to be verified and the material available to verify them is increasing, the presence of **automated tools that support fact-checkers** in identifying the relevant information and assist in writing a verdict for the claim can play an important role.

While the tool presented in the previous section focuses on reducing the amount of manual work by identifying which claims still need to be fact-checked, here the focus shifts to assist fact-checkers in issuing a verdict on the veracity of the claims. The **verdict generation tool** aims to support the creation of a verdict. This is done by exploiting the sources (articles) provided by the fact-checker to: i) identify within the articles the most relevant information for the claim; and ii) generate a verdict that explains the reasons for which the claim under investigation can be considered true, only partially true or false. To facilitate the sharing and dissemination of the verdicts, their generation is available with two different writing styles: **journalistic and social media styles**.

2.4.2. Related work

Previous work on automatic fact-checking involved **verdict prediction and justification production**. **Verdict prediction** is usually tackled as a classification task to determine the truthfulness of a claim. This classification task can be either interpreted as a binary "true/false" (Potthast et al., 2018; Popat et al., 2018) or multiclass to also include half-true claims (Wang, 2017), with some system focusing on evidence-based predictions (Thorne et al., 2018; Wadden et al., 2020). **Justification production** focuses on providing a rationale for the verdict. While some approaches are logic-based (Gad-Elrab et al., 2019; Ahmadi et al., 2019) or use attention-based techniques (Shu et al., 2019; Lu and Li, 2020), the most promising solution, at the moment, consists of **approaching justification production as a summarization task** (Kotonya and Toni, 2020; Atanasova et al., 2020; Guo et al., 2022).

2.4.3. Proposed method

The developed tool works by **receiving as input a claim and a text** (i.e., consisting of one or more articles). These will be **processed with a language model to generate a short verdict**, a few sentences long, explaining the reasons for which the claim can or cannot be considered true. As a design choice the verdict is generated based on articles that are provided by the users to **give fact-checkers the full control on the sources of information** they consider to be trustworthy. In addition, not using automatically-retrieved articles to generate a verdict helps to minimize errors on claims for which the veracity of the verdict is affected by variables such as date or location. Some claims can be true/false only when referring to a specific point in time or the very same claim can be true in a specific country and false in another one.

The **pipeline for verdict generation** follows the approach from Russo et al. (2023), combining extractive and abstractive summarization for justification production. In the **first step**, i.e., the extractive summarization, we start from the articles provided as input to identify the sentences that are more similar to the claim. To this end, SBERT (Reimers and Gurevych, 2019) is used to rank the sentence embeddings with respect to the claim using cosine-similarity. Then, in the **second step**, the claim concatenated with a reduced version of the input articles (obtained through extractive summarization) is provided as input to a transformer-based model[4] that was pre-trained on an abstractive summarization objective and fine-tuned on a dataset consisting in concatenated pairs

[4] Pegasus cnn_dailymail (Zhang et al., 2020)

of claims and reduced articles associated with a verdict. As a decoding mechanism to generate the verdict we rely on Top-p (nucleus) sampling with p = 0.9.

Datasets. The models used for English verdict generation are fine-tuned starting from 1,838 claim-article-verdict triples from FullFact website, and by using a subset of the LIAR-PLUS dataset (Alhindi et al., 2018) obtained by removing artificially created verdicts. To fine-tune the next version of the models for multilingual verdict generation, we are currently creating a multilingual dataset covering the languages involved in the project. In collaboration with the fact-checkers and the media partners of AI4TRUST (i.e., MALDITA, SkyTG24, DEMAGOG, EURACTIV, ADB, EMS), we are collecting claim-article-verdict triples in Romanian, Greek, Italian, French, English, Polish, Spanish, Hungarian and German.

2.4.4. Results and outlook

At the current stage, the tool is being tested on English data. We reproduced the results from Russo et al. (2023), and we report the observed performance in Table 2.4.1. The model is trained by combining both FullFact data and the subset of LIAR-PLUS, while the results on the respective test sets are reported separately. As a metric to evaluate the verdicts generated by the model, we adopt the ROUGE score (Lin, 2004) calculated between the generated verdict and the ground-truth. In Table 2.4.1, we report ROUGE-N (R1 and R2) which is based on the number of unigrams and bigrams overlapping, and ROUGE-L (RL) taking into account the lowest common subsequence between two texts.

	R1	R2	RL
LIAR-PLUS	0.473	0.261	0.370
FullFact	0.367	0.143	0.272

Table 2.4.1 F1 ROUGE scores of Pegasus cnn_dailymail fine-tuned on a unique dataset and tested on the LIAR-PLUS subset and FullFact test sets.

The **further development** of the tool has already started and is based on following two directions. In the **first one**, we are investigating the **impact of having multiple documents** as input on the identification of the relevant content. The results in Table 2.4.1 came from an approach that focused on generating a verdict from a single document; we are now performing preliminary tests to evaluate the performance of the models when receiving multiple articles and on using RAG (Retrieval Augmented Generation) to obtain the information relevant to the verdict. The **second direction** is extending the service to be **multilingual**, supporting all the languages of the project. For that, the data collection campaign is still ongoing.

2.4.5. Exposed API for integration

The verdict generation tool is made available through **REST APIs.** Given a claim and one or more articles as input, it will return: i) the top-*n* relevant sentences in the text with respect to the claim and ii) a textual verdict comprising the reasons why a claim can be considered as true or not. The service is still under development, but a **first version** of the API that supports the generation of verdicts in English, is **under testing.** The full documentation and entry point will be made available with the release of the first version of the tool. Below, we describe an example of the entry point and of the implementation with the required parameters.

Request parameters:

- **`claim`**: the input claim
- **`article`**: the input fact-checking article or articles
- **`sentences`:** the number of relevant sentences to be extracted from the text (e.g. 3)
- **`style`:** verdict writing style (journalistic/social)
- **`lang`**: the language of the input text (e.g. *en)*

Output:

- **`top_sentences`**: the most relevant sentences with respect to the provided claim extracted from the provided articles
- **verdict:** the text to be displayed as verdict

Below we show an example of the I/O, in JSON format.

Input:

```
{
   "claim": "Vaccines interfere with your DNA.";
   "article": "[...] The three Covid vaccines currently approved for emergency use in the UK have
demonstrably shown high levels of protection against infection, and the expected benefits of
the vaccines are said to “far outweigh any currently known side effects.” Some of the vaccines
used in the UK are mRNA vaccines.[...]";
   "sentences": "2";
   "style": "journalistic";
   "lang": "en"
}
```

Output:

```
{
   "top_sentences": [
         "Some of the vaccines used in the UK are mRNA vaccines.",
         "Vaccines themselves are extremely unlikely to weaken the immune system, and the
```

```
            benefit they have given to both humans and animals has been and continues to be
            enormous."
                                    ],
    "verdict": "There is no evidence to suggest that Covid-19 vaccines weaken or “degrade”
healthy immune systems. Vaccines themselves are extremely unlikely to weaken the immune
system in any way."
}
```

3. Audio data analysis methods

3.1. Speech-to-text

3.1.1. Problem statement

In AI4TRUST, **speech-to-text (STT)** is used to transcribe audio and video materials, such that their transcription can then be further processed with other text-based analysis tools. So, STT can be seen as a **pre-processing component of some of the text analysis technologies** described in Section 2. For example, classifying disinformation signals in textual content is one of the first tools that makes use of an STT module, to detect the prevalence of disinformation in various forms, such as hate speech, offensive language, and clickbait news. **Check-worthy claim detection** is another key component in the automated fact-checking pipeline, that will use as input the STT transcription, as it aims to detect texts presenting claims that are worthy of verification, that appear to be false, may be of public interest or of impact to the public, or may cause harm to the society, entities, groups, or individuals. These modules do not directly analyze the audio file, as they make use of NLP techniques, such as binary text classification, so the output of the STT module must have a low WER (word error rate) to achieve good overall performance. Moreover, STT is required in **many EU languages**, including English, Greek, Italian, Spanish, French, German, Polish, and Romanian. Overall, the STT component is crucial for the scalability of deepfake and disinformation detection framework, as it allows for fast processing of large-scale multimedia volumes. Furthermore, the integration of STT technologies enhances the capability to **dissect and understand the context and intent behind the content**.

3.1.2. Related work

The increased interest from the last decade in the field of automatic speech recognition resulted in the transition from research prototypes to mature technologies, successfully integrated in real applications. The massive adoption of deep neural networks facilitated the exploration of STT technologies, leading to the development of several cutting-edge tools and frameworks, specially designed **to enhance the accuracy and efficiency of converting spoken language into written text**.

Among these, Kaldi (Povey et al., 2011) is recognized for the transition from probabilistic algorithms to Time-delay neural networks (TDNN) (Peddinti et al., 2015) architectures. The STT systems created with this toolkit are hybrid systems, with an entire processing pipeline consisting of distinct components for feature extraction, acoustic modeling, and language modeling. Another important category of STT systems is those using the end-to-end approach, where a single neural network is responsible for the entire processing of the raw audio signal, up to the transcription. NeMo (Kuchaiev et al., 2019), developed by NVIDIA, incorporates a wide range of STT models based on end-to-end architectures, representing more complex variations of Convolutional Neural networks (CNN) and Recurrent Neural Networks (RNNs), such as Conformer-CTC, Conformer-Transducer (Gulati et al., 2020), LSTM-Transducer (Yanzhang et al., 2019) or Squeezeformer (Kim et al., 2022). Similar architectures are also available in Wenet (Yao et al., 2021), a toolkit that offers production-ready solutions for STT, being characterized by efficient models from the computational point of view, as well as real-time streaming speech recognition with remarkable accuracy.

Wav2vec (Baevski et al., 2020) from Facebook AI, employs a novel self-supervised learning approach where the model is trained to predict masked parts of the audio input, using a Transformer-based architecture. This technique has dramatically reduced the need for large amounts of labeled data, making STT training more accessible. OpenAI's Whisper (Radford et al., 2023) has made significant strides in STT by employing a large-scale Transformer model trained on a diverse range of internet-collected data, including various languages, accents, and noisy environments. Whisper is remarkable for its language detection capability and multilingual models.

3.1.3. Proposed method

During the AI4TRUST project, we updated our existing systems (hosted by the UPB partner) for STT in Romanian (Georgescu et al., 2021) and English, and created new systems for Spanish and Polish. Since we have extensive experience in using the Kaldi ASR toolkit, especially on Romanian language, we focused our initial efforts on improving results for these languages. For English, Spanish, and Polish languages, we established some **baselines models that can be further improved**. Furthermore, for the Romanian language, we also switched to a new toolkit, called NeMo. Lessons learned on NeMo during this phase of the project will later be transferred to the other languages.

For the **Romanian experiments**, we used the following 5 datasets for training: **BAS, SSC, COB, COR, and CDP**. The size of each dataset (expressed as the number of hours) and details on whether it was used to train different models, is reported in Table 3.1.3. BAS is our baseline training set, consisting of two approximately equal parts containing read speech and spontaneous speech. SSC is a set of spontaneous speech from radio and TV, automatically annotated. COB is another spontaneous speech corpus, created by automatic annotation starting from approximate transcriptions from excerpts and interviews. COR is more similar to BAS, while CDP was obtained by automatic annotation of recordings from Chamber of Deputies - Romanian Parliament. The evaluation corpora are manually annotated distinct subsets of the main training sets. RSC is the read speech part of BAS, while SSC1, SSC2 and CDP evaluation sets have the same composition as the corresponding training sets. More details about the datasets are available in Georgescu et al., 2021.

NeMo1 and NeMo2 are fast conformer transducer large models with greedy decoding, trained for 250 epochs, and the best model was obtained by averaging the best 5 and 10 checkpoints, respectively. For **data augmentation**, we used augmentations such as speech rate and additive noise perturbations. **NeMo3** is a slight variant of NeMo2 that achieved the best result on SSC1. **NeMo4 and NeMo5** are variations of fast conformer hybrid large models with greedy decoding, trained for 50 epochs with speech rate and noise augmentation. Additionally, they also use a 6-gram language model. Therefore, by using additional training data and NeMo, we observed significant relative improvements compared to the baseline on spontaneous speech, namely around 14% on SSC1 and SSC2 and 35% on CDP.

For the **English, Polish and Spanish languages**, our approach consists in training chain **TDNN models** (Povey et al., 2016) using the Kaldi toolkit. The TDNN architecture was selected for its capability in capturing large temporal context and its efficiency in dealing with variable input lengths; both characteristics are crucial for handling a wide range of spoken nuances. TDNNs, by design, can model long-term dependencies in speech signals better than traditional neural networks, thanks to their specialized structure that applies time-delay layers to process sequential data over different time scales.

In order to train the **English model**, we used the **LibriSpeech corpus** (Panayotov et al., 2015), a publicly available dataset specifically designed for speech recognition research. The LibriSpeech corpus is derived from audiobooks as part of the LibriVox project, providing a high variety of content. It comprises over 1,000 hours of spoken English including novels, short stories, and poetry. During the training stage, we focused on hybrid HMM-TDNN acoustic models. The input of this network is composed of 40-dimensional mel-cepstral features (MFCCs) and 100-dimensional i-vectors. Regarding language modeling, we use a probabilistic 4-gram model at decoding and an RNN-LM (Mikolov et al., 2010) for rescoring.

For **Spanish**, we based our Kaldi recipe on datasets mostly from the **Heroico corpus**[5]. The Heroico corpus was originally collected to train acoustic models for pronunciation modeling in Spanish language learning applications. The corpus consists of two main sub corpora: a) a subcorpus collected at Mexico's Military Academy called Heroico, and b) a subcorpus collected at the United States Military Academy (USMA) in West Point New York. The Heroico corpus is further divided into recited and prompted speech subcorpora. The USMA subcorpus includes 1.2 hours of speech from nonnative speakers and 1 hour of speech from native speakers. All the speech in the USMA corpus was recited. Except for one hour of speech, the Heroico subcorpus was used for training and the USMA subcorpus was used for evaluation. The Heroico subcorpus has 11.8 hours of speech. We used 10.8 hours as training data. 2.2 hours of speech from the USMA subcorpus was used as evaluation data. One hour segment of speech in the Heroico corpus was recited from the same set of prompts that was used in the USMA collection. To avoid overlap of the training and evaluation

[5] https://www.openslr.org/39/

sets, this one-hour segment was separated out from the Heroico corpus into an evaluation set. As input features, we used the standard 40-dimensional mel-cepstral features (MFCCs) and 100-dimensional i-vectors, similar to the English recipe.

For **Polish language**, we based our recipe on **Clarin-PL project**[6]. This project, conducted at the Polish consortium of the CLARIN project (Koržinek et al., 2017), seems to provide one of the largest high-quality studio speech corpora, released under an open license, for the area of Polish speech research. The corpus consists of 317 speakers recorded in 554 sessions, where each session consists of 20 read sentences and 10 phonetically rich words. The size of the audio portion of the corpus amounts to around 56 hours. The corpus is split into a training and evaluation portion, roughly 90% and 10%, respectively. A trigram statistical language model is used, provided by the same project. The time-delay neural network (TDNN) system in Kaldi achieves the best score, as we have seen with previous languages.

3.1.4. Results and outlook

For **performance evaluation**, we used Word Error Rate (WER)[7] metric, as it is a common metric used to evaluate the performance of automatic speech recognition (ASR) systems. The Word Error Rate (in percentage) is calculated as follows: %WER = (Substitutions + Insertions + Deletions) / Total Number of Reference Words. The WER provides a measure of how well the ASR system has performed in transcribing the speech input. Typically, a WER below 10-15% is considered good performance for most practical applications.

For **Romanian language**, the first two lines in Table 3.1.3 represent the baseline systems reported by us in the past in (Georgescu et al., 2021). Both systems are trained with Kaldi, TDNN-HMM acoustic models, with a 4-gram language model for decoding and RNN for rescoring. The results on read speech are the best, but this is not so relevant, given the fact that RSC-eval is a somewhat artificial evaluation set, recorded in the laboratory, without noise and with good diction. Instead, we are more interested in spontaneous speech datasets, represented by speech recorded in real situations, and we used NeMo in an attempt to achieve this goal. Therefore, by using additional training data and by using the NeMo Toolkit, significant relative improvements were obtained compared to baseline on spontaneous speech; around 14% on SSC1 and SSC2 and 35% on CDP, to obtain a WER as low as 3.96% on NeMo2 system, on CDP dataset.

For **English**, the Librispeech dataset is a widely used benchmark for speech recognition in this language. As such, multiple recipes and setups are available to obtain the lowest WER possible on this corpus. Furthermore, we chose to complement our evaluation set with evaluation sets from

[6] https://github.com/danijel3/ClarinStudioKaldi/tree/master
[7] https://en.wikipedia.org/wiki/Word_error_rate

TEDLIUM. TEDLIUM dataset is another widely used benchmark for speech recognition in the English language. It is derived from TED Talks, which are a series of influential and informative talks on a wide range of topics, so it is spontaneous speech. In our pipeline, we managed to achieve a 5.69% WER on librispeech test-clean set and 6.05 tedlium.

System ID	Training set					Evaluation set (WER[%])			
	BAS 225h	SSC 292h	COB 31h	COR 84h	CDP 2049h	RSC 5.5h	SSC1 3.5h	SSC2 1.5h	CDP 5.0h
Kaldi1	✓					1.88	14.96	20.00	10.80
Kaldi2	✓	✓	✓	✓		**1.60**	10.30	12.20	6.10
NeMo1	✓	✓	✓	✓		3.68	10.69	12.02	8.70
NeMo2	✓	✓	✓	✓	✓	3.13	9.84	12.07	**3.96**
NeMo3	✓	✓	✓	✓	✓	3.05	9.28	**10.50**	4.12
NeMo4	✓	✓	✓	✓	✓	-	9.13	11.31	-
NeMo5	✓	✓	✓	✓	✓	-	**8.92**	10.91	-

Table 3.1.3. Comparison between baseline STT systems for Romanian language (Kaldi1&2) and newly trained NeMo based systems.

For **Spanish**, we used the TDNN-F architecture (Factorized Time Delay Neural Network), with a total number of 8 training epochs. For the language model, we used the Santiago Spanish Lexicon dataset[8]. We trained the model on Heroico answers and Heroico recited sets. The resulting model obtains WERs varying between 6.13% for native speech (USMA native set), 12.47% for non-native speech (USMA non-native set) and 9.64% for recited speech (Heroica recited set).

Finally, for **Polish**, the time-delay neural network (TDNN) system achieves, as expected, the lowest WER, significantly better than GMMs and the LSTM architectures we tested, to obtain the lowest WER of 5.91%, with a TDNN architecture and large LM rescoring (4-gram language model).

As future steps, we **plan to use fast conformer transducer models** for languages besides Romanian, as the results presented above show that such models obtain better results. In addition, we plan to create new models for the additional languages envisaged by the AI4TRUST project, namely **Italian, German, and French**. All these models will be evaluated and possibly fine-tuned on the so-called **“Ground Truth” datasets developed in WP2**.

[8] https://openslr.trmal.net/resources/34/

3.1.5. Exposed API for integration

Our **API** is **REST-based** and is built from scratch in Java. It offers Speech-to-text transcription services, for a given multimedia file. We use two endpoints, one for media upload and the second for requesting the transcription results.

The upload POST endpoint accepts a JSON object, exemplified below:

```
Endpoint: https://transcriptions.speed.pub.ro/GatewayAPI/media/transcription
```

```
{
  "api_key": "your_api_key",
  "asr_domain": "en-US_general",
  "content_type": "URL",
  "content": "https://www.youtube.com/watch?v=ZHfPp-NBkG0"
}
```

Supported ASR domains are ro-RO_general, en-US_general, for now. Spanish and Polish languages will be added soon. Content_type object can be a string named “URL” or “base64”, to indicate if we need to fetch an external URL (for Facebook or Youtube). If the “base64” option is sent, then the “content” field must be a base64 encoded audio file.

In order to fetch the final transcription, we offer a second POST endpoint to fetch your results, exemplified below:

```
Endpoint:
https://transcriptions.speed.pub.ro/GatewayAPI/transcription
```

```
{
  "api_key": "your_api_key",
  "job_id": "12367”
}
```

If successful, it will output the transcription JSON with timestamps (please find a partial example below):

```
{
…
  {
    "word": "chaotic",
    "confidence": 0.99,
    "start": 189.02,
    "length": 0.15
  },
  {
    "word": "time",
    "confidence": 0.98,
    "start": 189.18001,
    "length": 0.53
  }
],
"transcript": "i i think the summer i think that anyone who follows the legend
like next day is going to have a tough time we've seen this route sports
history where the person who comes in right after the great coach has some
trouble obviously fans get disgruntled quickly if that person doesn't emulate
the success of the previous coach their legendary start like the bear bryant
alabama years ago it was very tough to replace him until they found nickname
they couple of others that were very good but it takes a while and so it is
going to be very interesting to see especially with the if he can recruit
recruit the young players the high school players the way that nick saban was
so successful like a magnet bringing them to alabama and especially now for
the college game with name image and likeness with the transfer portal all the
upheaval there's conference realignment is a very very chaotic time"
}
],
"final": true
…
}
```

3.2. Deepfake audio detection

3.2.1. Problem statement

Deepfake audio detection aims to predict whether a given audio is real (*bonafide*) or synthetically generated (fake or spoofed). This task attempts to prevent the misuse of speech synthesis technology, which has been rapidly improving (Liu et al., 2023, Kim et al., 2023, Masood et al., 2023, *inter alia*) and can produce high-fidelity voice clones from only a few seconds of speech.[9] While there has been a sustained effort on the task of deepfake detection (see the following Section 3.2.2), many of the ensuing methods lack **two critical properties** for real-world use, that relate to the **generalizability** (i.e., the ability to perform well on new data that have not been used at training) and **trustworthiness** (i.e., the capacity to be reliable or well calibrated). Here, we present our approach towards these goals.

3.2.2. Related work

Audio deepfake detection is a **very active field of research**, showcasing many promising results (Tak et al., 2021; Wang et al., 2022; Zhang et al., 2023). However, Müller et al. (2022) have shown that many of these methods fail to perform well on out-of-domain realistic samples. A possible explanation for the poor generalization is the applied preprocessing to the training dataset (ASVspoof'19; Wang et al., 2020): silence duration (Müller et al., 2021) and bitrate information

[9] https://elevenlabs.io/voice-cloning (accessed: 2024-03-22)

(Borzì et al., 2022) correlate with the “Ground Truth” datasets of WP2. Given that the best deepfake detection models are high capacity, they rely on these low-level spurious features. As a remedy, Ojha et al. (2023) proposed the use of frozen self-supervised representations and showed that these representations offer a much better generalization capacity. While these experiments were carried out in the context of images, a similar line of research emerged also in the speech community (Wang et al., 2022; Tak et al., 2022). However, these approaches either used smaller representations with modest results or they did not focus on the generalization aspect (they were tested in-domain and not on other challenging datasets, such as the “In the Wild” dataset (Müller et al., 2021)). Other works fine-tuned these features or integrated them into more complex systems (Wang et al., 2022; Xie et al., 2023), but in this way, they lose the implicit generalization power.

3.2.3. Proposed method

The developed method first extracts speech representations using pretrained (frozen) self-supervised models and then trains a linear binary classifier (logistic regression) on top of these representations. Since we learn a simple linear layer, **our method avoids overfitting and enables generalization**. Since logistic regression estimates probabilities, we obtain **better-calibrated estimates than deep learning algorithms**, which are typically overconfident (Guo et al., 2017).

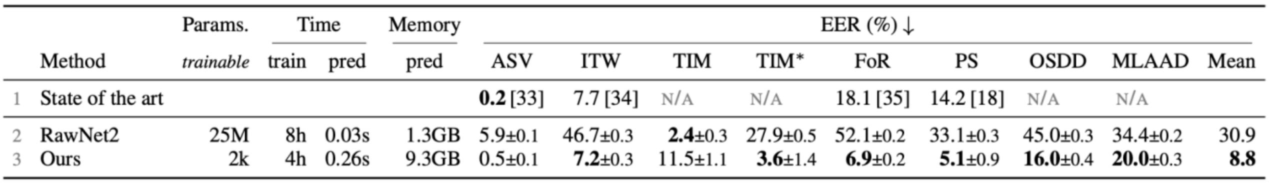
Self-supervised representations. We use self-supervised representations stemming from the wav2vec 2.0 (Baevski et al., 2020) method. We have chosen this family of models because it has proven strong transfer abilities and comes in multiple variants, enabling us to assess the importance of various factors, such as model size or pre-training data. Wav2vec performs unsupervised pre-training on raw audio data and, as a result, learns useful speech representations without the need for annotations. Apart from the original wav2vec method, we use two of its extensions: XLS-R (Conneau et al., 2021), which learns cross-lingual representations, and WavLM (Chen et al., 2022a), which considers the task of speech denoising in addition to the masked audio prediction task in wav2vec.

Calibration and reliability estimation. A classifier is calibrated if its predictions match the accuracy obtained for that particular level of confidence. We apply the logistic regression classifier, which uses the cross-entropy loss. The cross-entropy loss is a proper loss (Błasiok et al., 2023), which improves the calibration properties. This choice avoids the need for other post-processing techniques such as Platt's scaling (Platt, 2000). Calibration is also related to generalization: Carrell et al. (2022) have shown that the calibration error is bounded by the generalization error. Calibrated probabilities help with related downstream tasks (Bhatt et al., 2021). Here, we consider reliability estimation, which is useful for rejecting examples for which the model is unsure. Given the estimated probability of the audio being fake, we obtain uncertainty estimates using the entropy.

3.2.4. Results and outlook

How well do self-supervised representations generalize?

We measure how well our method generalizes by training on the ASVspoof'19 dataset (Wang et al., 2020) and evaluating on a benchmark of eight different datasets, including partially spoofed and multilingual datasets. The results of the conducted experiments are shown in Table 3.2.1. We contrast the performance of pretrained self-supervised representations (rows 3) to that of the best models in the literature (rows 1) and the RawNet2 model (row 2), one of the best models on generalization according to Müller et al. (2022). As the self-supervised representation, we use the 2B XLS-R variant from wav2vec 2.0, `wav2vec2/xls-r-2b`, which is the largest model and trained on most data.



		Params.	Time		Memory	EER (%) ↓								
	Method	*trainable*	train	pred	pred	ASV	ITW	TIM	TIM*	FoR	PS	OSDD	MLAAD	Mean
1	State of the art					**0.2** [33]	7.7 [34]	N/A	N/A	18.1 [35]	14.2 [18]	N/A	N/A	
2	RawNet2	25M	8h	0.03s	1.3GB	5.9±0.1	46.7±0.3	**2.4**±0.3	27.9±0.5	52.1±0.2	33.1±0.3	45.0±0.3	34.4±0.2	30.9
3	Ours	2k	4h	0.26s	9.3GB	0.5±0.1	**7.2**±0.3	11.5±1.1	**3.6**±1.4	**6.9**±0.2	**5.1**±0.9	**16.0**±0.4	**20.0**±0.3	**8.8**

Table 3.2.1. Comparison in terms of equal error rate (ERR) with state-of-the-art on multiple out-of-domain datasets.

We observe that pretrained representations perform on average much better than RawNet2: 8.8% versus 30.9% equal error rate (EER). The performance is also better on each dataset with a single exception. Our method also compares favorably to many of the state-of-the-art methods. These are much more complex approaches, which are evaluated only on a handful of datasets. For several datasets, we are the first to either report results (OSDD) or the first to report results in terms of the chosen metric, EER (TIM, TIM*, MLAAD).

The computational cost of self-supervised representations is dominated by the feature extraction step. Processing an audio of 3 seconds on a Tesla T4 GPU takes around 0.3 seconds with a video memory consumption of around 9 GB. These requirements are an order of magnitude larger than those of RawNet2, but still reasonable in the absolute and attainable by commodity hardware.

How reliable are the self-supervised representations?

We assess whether we can trust the predictions of deepfake detectors or not. Following Salvi et al. (2023), we formulate this desideratum as the task of reliability estimation: we want the model to be able to assess the level of confidence in its predictions; a high confidence indicates a high probability that the prediction is correct. To this end, we encode the confidence in a prediction using the entropy of the generated probabilities (see Section 3.2.3): if the entropy is close to zero, the model deems the prediction to be reliable; conversely, high entropy indicates uncertain inputs.

We use two metrics to measure the reliability estimation capabilities (Nadeem et al., 2009): the fraction of data that is reliable and the classifier accuracy on this data. We produce curves by varying a threshold τ from 0 to 1 in steps of 0.01 on the unit-scaled entropy of each prediction.

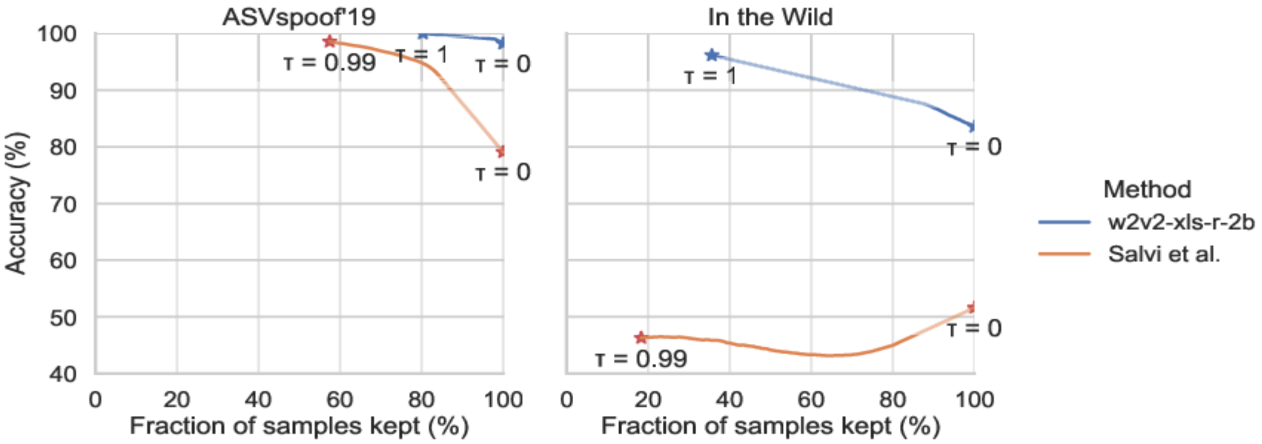


Figure 3.2.1. Evaluation of reliability estimation in terms of accuracy and fraction of samples kept, as we vary the reliability threshold $\tau \in [0,1]$.

We report results on the ASVspoof'19 and “In the Wild” datasets, The results are shown in Figure 3.2.1. We observe much better results than prior work on both datasets, in terms of both metrics and at all thresholds. Naturally, there is a drop in performance when going out-of-domain (on the “In the Wild” dataset), but it is less severe than what we observe for the method of Salvi et al., (2023). Moreover, on the “In the Wild” dataset we see that we can trade off data kept for accuracy, which is not the case for the other method.

How do other self-supervised representations perform?

Pre-trained self-supervised representations come in multiple variants, differing in terms of architecture, model size, or pre-training data. Based on this information we decouple the performance on three axes: model family, model size, data. We analyze 11 variations of self-supervised representations (see Section 3.2.3) in terms of both discrimination and calibration error on four out of the eight datasets previously used.

We summarize our findings as follows: i) larger self-supervised models perform better: the mean error decreases from 16.2% to 6.6% when increasing the model size from 300M to 2B parameters; ii) XLS-R is the better wav2vec model family: the 300M-parameter XLS-R model yields 16.2% mean error versus 24.7%, obtained by the WavLM counterpart; iii) pre-training data influences results, but it remains difficult to conclude what type of data helps - sometimes models trained on more data perform worse than models trained on less data; and iv) the conclusions above hold also for the calibration performance.

How important is the classifier?

We have experimented with two more flexible models on top of the frozen self-supervised representations: a three-layer multilayer perceptron with ReLU activation, and a self-attention layer followed by a linear layer. Additionally, we have investigated a stronger regularization value for logistic regression. On average, logistic regression obtains the best results on both classification and calibration, with the less regularized variant performing better (the logistic model, being linear, is already highly constrained, so further regularization results in underfitting).

Outlook

So far, we have proposed a simple yet effective method for audio deepfake detection. Our method relies on frozen self-supervised features (the 2B-parameter wav2vec 2.0 XLS-R model being the best-performing variant) and learns a single layer via logistic regression. As empirically demonstrated, this approach enhances generalization and offers more trustworthy predictions that encode the model's uncertainty. This method is also fairly efficient from a computational standpoint running at 10× real time, with the bottleneck residing in the feature extraction step.

Going forward, we plan to **evaluate the audio deep fake detection method in a real-world scenario** using real and fake audio data distributed in social media. Preliminary results show that the proposed method does not generalize well on such data. We, therefore, need to assess if more data or rather more complex methods are required to correctly identify real-world deepfakes.

3.2.5. Exposed API for integration

Our **API** is **REST-based** and is built from scratch in Java. The integrated audio deepfake detection method analyzes the waveform of the submitted media file, while the duration of the analysis ranges from 40% to 100% of the media's duration. We use two endpoints, one for media upload and the second for requesting the deepfake score. The upload POST endpoint accepts a JSON object, exemplified below:

```
Endpoint:
https://transcriptions.speed.pub.ro/GatewayAPI/media/deep-fake
```

```
{
    "api_key": "your_api_key",
    "content_type": "URL",
    "content": "https://www.youtube.com/watch?v=ZHfPp-NBkG0"
}
```

Where *content_type* can be a string named "URL" or "base64", to indicate if we need to fetch an external URL (for example Facebook or Youtube). If the "base64" option is sent, then the "content" field must be a *base64*-encoded wav audio file. For fetching videos from external URLs, we use youtube-dl repo, which supports fetching from an extensive list of websites/applications: https://github.com/ytdl-org/youtube-dl. In order to fetch the final deep fake detection score, we offer a second POST endpoint to fetch your results, exemplified below:

```
Endpoint:
https://transcriptions.speed.pub.ro/GatewayAPI/deep-fake
```

```
{
  "api_key": "your_api_key",
  "job_id": "12367”
}
```

If successful, it will output the detection score (a number between 0 and 1, where 1 is 100% deep-face probability):

```
{
  "deep-fake-score": "0.89444872956905"
}
```

3.3. Deepfake audio generation

3.3.1. Problem statement

In order to thoroughly test the developed method for deepfake audio detection (as well as our method for audio anomaly detection that is described in Section 5.2), we need to be also able to **generate audio samples** from the latest high-quality technologies and methods, as well as to **understand how these methods are able to trick the deepfake detector**. While the field of text-to-speech synthesis (TTS) and voice cloning (VC) is extremely vast and rapidly advancing, we are not aiming to use all of the available methods, but rather to find how different architectural choices reflect or affect the final output speech sample. We are also addressing **the problems of fast speaker adaptation**, also in cross-lingual scenarios, and audio inpainting (or partial generation).

Within the current landscape of deepfake generators, the latest generation of text-to-speech synthesis systems have **increased the probability of mislabeling fake samples as real.** One of the reasons is the fact that these systems are now able to mimic a speaker's voice from as little as a few seconds of an audio recording. This mimicking is to some extent limited, meaning that for very short utterances, it may be hard to distinguish between the original and the fake speech. However, when the length of the utterances increases, certain speaker traits, such as rhythm, inflections, verbal ticks etc., may not be as easily copied. Still, their malicious use is easily attainable, and needs to be addressed in a coherent, targeted manner.

Also, deepfake detection is at the moment mostly addressing the use of completely generated samples. However, perpetrators may cleverly design and use partially generated samples, where only a word or a sequence of words may be replaced, removed or inserted, thus changing the intended message. In this case, **a text-to-speech synthesis system should also consider the**

context within which the generation occurs. This is a less studied problem, very similar to image inpainting, yet still in need of more research endeavors. Understanding how such methods may be designed would lead us to understand how the countermeasures can take **effect in real life applications**.

3.3.2. Related work

Text-to-speech synthesis has seen a very large increase in interest over the last 5-10 years, mostly as a result of the fast and high-quality deployment of **deep neural networks**-based architectures. The architectures' ability to provide natural sounding samples with a limited number of samples, as well as their ability to perform knowledge transfer from one speaker to another, even across languages, makes them a very easy to use tool which enables high-quality synthetic speech generation. The latest TTS architectures follow the general **generative AI research directions** and are mostly based on diffusion principles. The common ideas across the proposed methods involve the use of additional conditioning by factoring speech into duration, pitch, and speaker representations (Ju et al, 2024; Liu et al., 2022; Kim et al., 2022) which stem from the transformer-based model FastSpeech2.

The **stability and optimisation of the inference speed and complexity** is also addressed, with some of the most notable methods mitigating the sampling drift (Xue et al., 2023), residual prediction (Chen et al., 2022b) or progressive distillation (Vovk et al,, 2022). On the fast adaptation issue, the methods of (Liu et al., 2023b; Kumar et al., 2022; Yihan et al., 2022) propose the use of efficient speaker representations which enable complexity and data optimized target speaker adaptation. A notable approach is that of UnitSpeech (Kim et al., 2023). The architecture is based on the Grad-TTS (Popov et al., 2021) diffusion model, yet it removes the textual transcription dependency for adaptation by using HuBERT-derived codes. These codes are used as an **alternative encoder module** when the audio transcripts are not available for the target speaker.

3.3.3. Proposed method

Off-the-shelf pretrained models

To extend our spoofed samples library, we first resorted to using the **available pre-trained models** from some of the latest stable TTS architectures: FastPitch,[10] GradTTS,[11] UnitSpeech,[12]

[10] https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/tts_en_fastpitch
[11] https://github.com/huawei-noah/Speech-Backbones/blob/main/Grad-TTS/README.md
[12] https://github.com/gmltmd789/UnitSpeech

CoMoSpeech,[13] and RadTTS.[14] The models included both single speaker, as well as multiple speaker systems (if available).

Romanian TTS systems

For three of the most promising architectures, we also analyzed their potential for fast deployment, retraining and adaptation across multiple languages. We selected the FastPitch and Grad-TTS architectures, so far, and we are now experimenting with CoMoSpeech. For these systems, we evaluated their results on a multispeaker Romanian dataset, and found that similar quality and training/inference speed can be obtained as their English counterparts.

High-quality partially spoofed samples

A separate task that we envisioned for the use of TTS systems, was that of creating a new extended, high-quality, partially spoofed audio dataset. At that time, the only such datasets were PartialSpoof (Zhang et al., 2022)[15] and LAV-DF (Cai et al, 2022).[16] The PartialSpoof dataset contains samples from the rather old ASVSpoof19 resource, thus including rather poor quality TTS and VC samples. It is also constructed such that some of the samples contain a mix of real audio from different speakers. These samples are useful for audio splicing evaluation, yet not in our focus at this point in the project. The LAV-DF dataset, on the other hand, is an important resource as it also includes fake videos–making it useful even for multimodal deepfake detection. The dataset is built by substituting/removing/inserting a single word or syntagm. However, we found that some of these manipulations were far from accurate, and artifacts were clearly audible.

As a result, we attempted to recreate the LAV-DF dataset using UnitSpeech, while also investigating better video generation methods. For each of the fake samples in LAV-DF, we adapted the baseline UnitSpeech model to the target speaker. The entire utterance was then generated from the available transcripts, and the fake segment was copy-pasted within the original real carrier utterance. A first evaluation of this process revealed a major improvement in the audio quality over the original LAV-DF data.

However, while working on the new dataset, the LAV-DF authors released an updated dataset, called AV-DeepFake1M (Cai et al, 2023) with over 1 million videos. We contacted the authors and obtained access to the data for a general evaluation. Their results were clearly superior to the previous dataset, and on par with our UnitSpeech generated audio data. We therefore abandoned the generation step, but maintained the audio samples which will be used for further internal evaluation of our deepfake detection systems.

[13] https://github.com/zhenye234/CoMoSpeech
[14] https://github.com/NVIDIA/radtts
[15] https://zenodo.org/records/4817532
[16] https://huggingface.co/datasets/ControlNet/LAV-DF

3.3.4. Results and outlook

At the moment, the results of the tested TTS and VC algorithms have mostly been used for **internal testing of our deepfake detection methods.** Over 10k samples have been generated with the previously mentioned methods, mostly in **English and Romanian**. If these samples can be used to augment existing deepfake detection datasets (such as that ones developed by our sister project, i.e. Yaroshchuk et al., 2023), we will take all the necessary actions to prepare and include them.

As following steps, we will target the **generation of fake samples in multiple languages**. We have already started to retrain some of the architectures using data in **German, Spanish, and Italian.** On a separate track, we will also attempt to **combine all the trained text-to-mel models with various neural and signal-based vocoders**, as these may exhibit common artifacts irrespective of the underlying TTS architecture. A constant endeavor in the TTS and VC research fields is, of course, that **of fast adaptation to new target speakers** using zero- or one-shot approaches. Following the evaluation of the existing methods, we will also explore such tasks in **multilingual speech synthesis systems.**

3.3.5. Exposed API for integration

For the moment this technology is not exposed through an API. Moreover, we do not foresee integrating this technology in the AI4TRUST platform. Consequently, **exposing this technology through an API will not be necessary.**

4. Visual data analysis methods

4.1. Reverse video search on the Web

This section reports our work on **extending an existing web-based tool for video fragmentation and reverse search** on the Web (developed by CERTH in the past), to **automate interaction with online search engines** (e.g., Google) and **facilitate further the detection of near-duplicates of a query video** on the Web. The easier detection of such duplicates will, subsequently, expedite the debunking of disinformation that relies on the re-use of an older video to mislead the viewers about a recent or ongoing event.

4.1.1. Problem statement

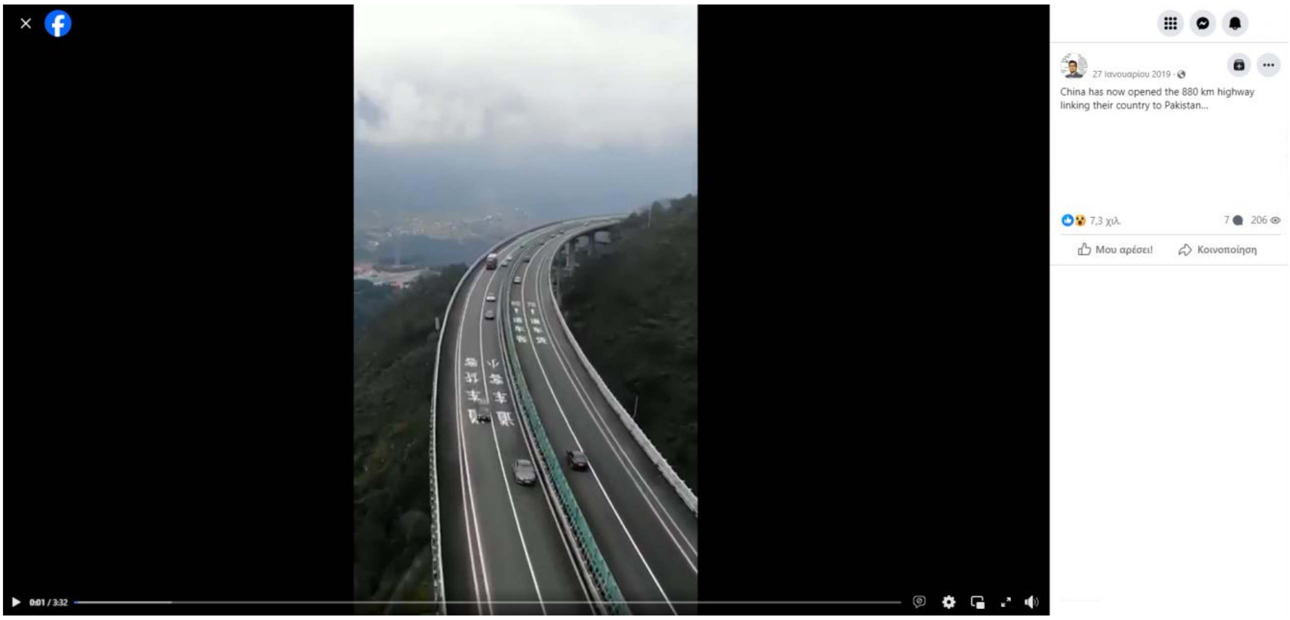
One type of disinformation is based on the **reuse of a video** from an earlier event to mislead the viewers about a recent or even ongoing event. Such **fakes** are probably the easiest to do and spread via **social networks and video sharing platforms**, and thus are the most commonly found by journalists and fact-checkers[17]. An example of such a fake that went viral on social media, is depicted in Fig. 4.1.1. According to the associated claim, China opened an 880 km-long highway which connects China with Pakistan. Almost four minutes long, the video shows an aerial view of a swanky elevated corridor, with several vehicles passing through it. The “highway” passes through several terrains, right from mountains laden with snow to a river beneath it. After being posted on social media, the video has received close to 2M views. However, the original video that was posted online a few weeks before the fake claim, shows the Yaxi Highway, a 240 Km-long highway which connects Ya'an and Xichang in southwest China's Sichuan province (see Fig. 4.1.2).

The **identification and debunking of such fakes** require the detection of prior occurrences of this video (or parts of it) on the Web, in order to trace the original story/event behind the video. A baseline approach for performing this task requires the user to manually take screenshots of the video in the player and use them to perform image-based search using the corresponding functionality of the most popular web search engines (e.g., Google search, Bing, Yandex, Baidu). Nevertheless, this process can be highly laborious and time-demanding, while its effectiveness depends on a limited set of manually taken screenshots of the video. A more advanced approach is supported by **the “Keyframes” component of the "Fake News Debunker by InVID & WeVerify" extension**[18] for Chrome. This component allows the users to process a video, extract a set of representative keyframes and use them to perform image-based search using a variety of search engines, in order to find near duplicates of the video on the Web. The technology behind the

[17] Indicative examples of recently debunked fakes that rely on video re-use, can be found at: https://news.google.com/search?q=%22fact%20check%22%20%22keyframes%22&hl=en-US&gl=US&ceid=US%3Aen

[18] https://chromewebstore.google.com/detail/mhccpoafgdgbhnjfhkcmgknndkeenfhe

“Keyframes” component was developed by the AI4TRUST partner CERTH, while its functionality is supported also by a standalone web-based tool[19]. Such a tool can significantly facilitate the detection of near-duplicates of a given video on the Web, through the provision of a rich set of representative keyframes and the supported interaction between the user and the search engines. However, the amount of this interaction in some cases can be high, as the user might need to repeat the search using multiple keyframes and multiple search engines. So, we argue that a more automated use of the video processing results (i.e., the extracted video keyframes) for searching the Web, would minimize the amount of required user interaction and would further **facilitate the retrieval of near-duplicates** of the processed video, from the Web.

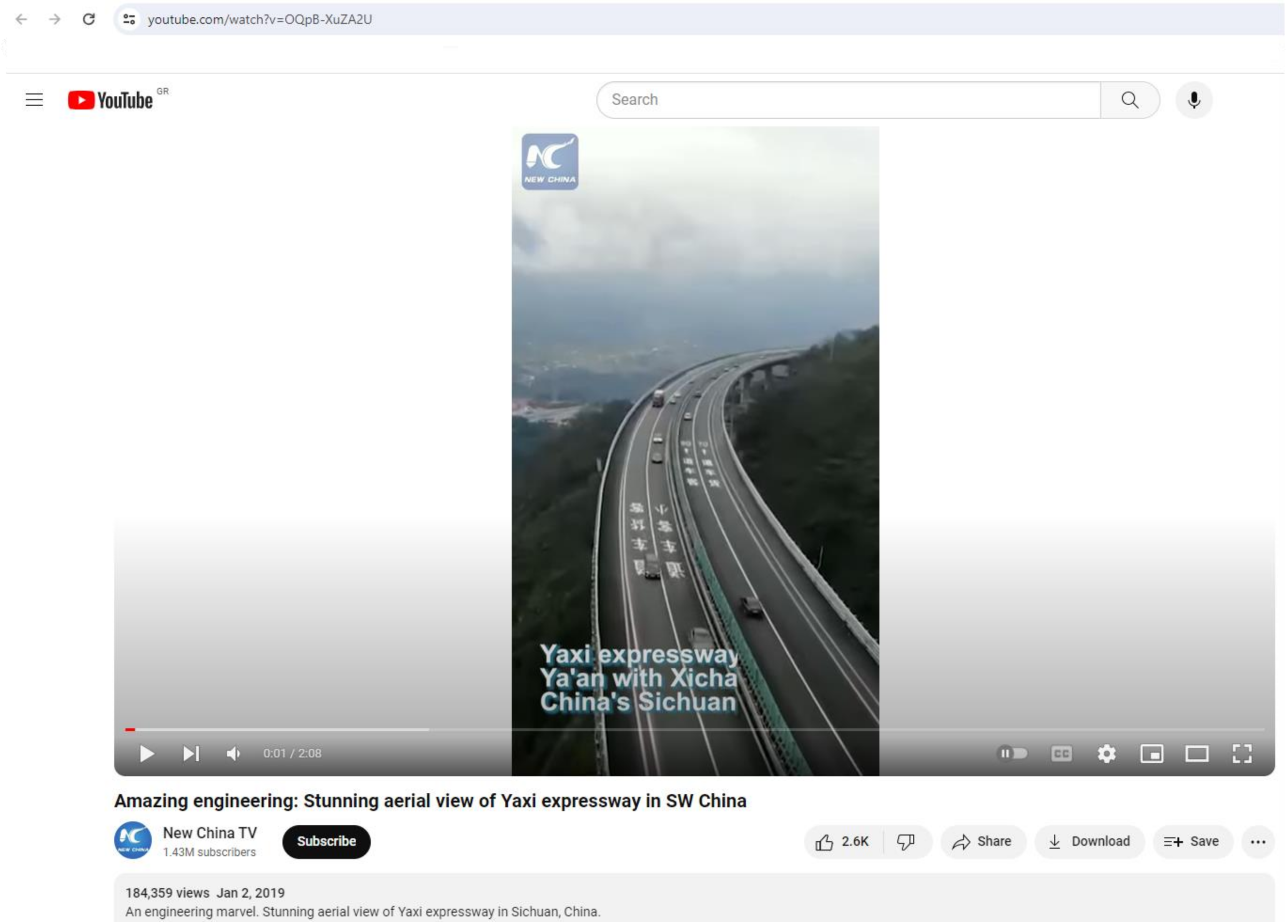


27 Ιανουαρίου 2019 ·

China has now opened the 880 km highway linking their country to Pakistan...

Fig. 4.1.1. A fake video posted on social media, claiming that China opened an 880 km-long highway which connects China with Pakistan.

[19] Accessible at: https://multimedia3.iti.gr/video_fragmentation/service/start.html



Amazing engineering: Stunning aerial view of Yaxi expressway in SW China

New China TV
1.43M subscribers

184,359 views Jan 2, 2019
An engineering marvel. Stunning aerial view of Yaxi expressway in Sichuan, China.

Fig. 4.1.2. The original video, showing the Yaxi Highway in southwest China's Sichuan province.

4.1.2. Related work

Several tools were introduced over the last years, to support **reverse search on the Web using visual data.** A couple of them (TinEye[20] and RevEye[21]) allow the user to perform reverse search on still images using the corresponding functionality of online search engines, while other technologies (Berify[22] and Videntifier[23]) enable this reverse search only within closed collections of images and videos, thus significantly restricting the boundaries of investigation. The DataViewer of Amnesty International[24] extends the online searching capability of the aforementioned solutions, by supporting the reverse search of YouTube videos using a (restricted) set of video thumbnails for reverse image search. The current state of the art on video reverse search on the Web is represented by technologies such as the “Keyframes” component of the "Fake News Debunker by InVID & WeVerify" extension for Chrome. This technology facilitates the detection of near-

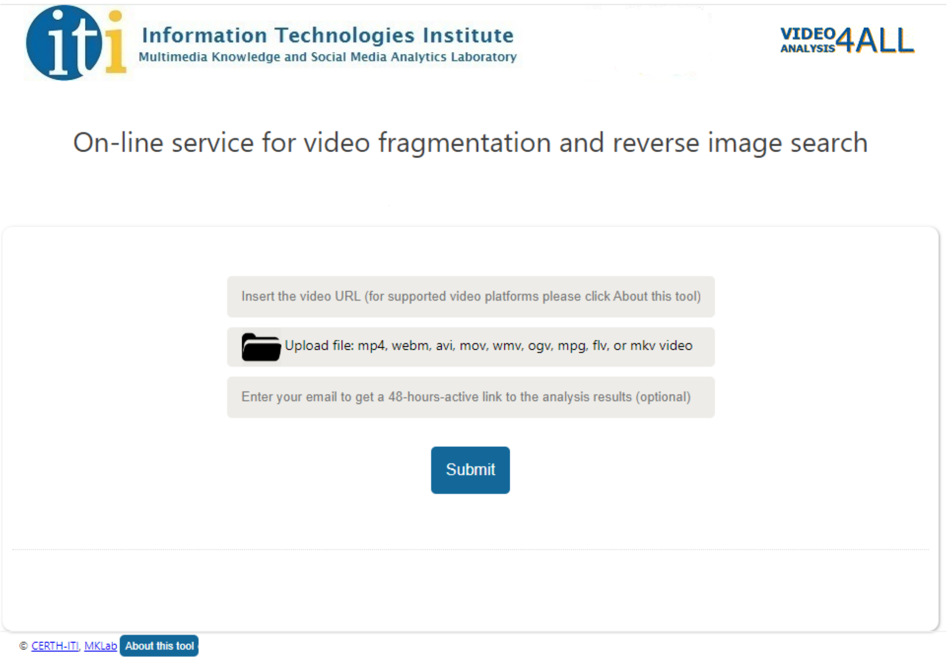
[20] https://tineye.com/
[21] https://chromewebstore.google.com/detail/reveye-reverse-image-sear/keaaclcjhehbbapnphnmpiklalfhelgf
[22] https://berify.com/
[23] https://www.videntifier.com/
[24] https://citizenevidence.amnestyusa.org/

duplicates of a given video on the Web, by extracting a set of representative video keyframes and allowing the user to perform reverse search on the Web using one or more selected keyframes and multiple search engines. In terms of compatibility, the supported on-line video sources include YouTube, Facebook, Twitter, Instagram, Vimeo, DailyMotion, LiveLeak and Dropbox (though, not all videos from these platforms are accessible due to platform-specific or user-defined restrictions about the use of each specific video), the supported video formats include: mp4, webm, avi, mov, wmv, ogv, mpg, flv and mkv, and the supported search engines include Google, Bing, Yandex, TinEye, Baidu and Reddit. Thus, this tool allows the user to analyze videos from the most widely used platforms and perform an exhaustive investigation through multiple search engines. However, the reverse video search is currently performed via a semi–automated process that requires the selection of one or more keyframes and the interaction of the user with the searching results. In AI4TRUST, we aim to further facilitate this process by automating the interaction between the user and the search engines, thus minimizing the amount of needed user intervention.

4.1.3. Proposed method

We built on the **web-based tool for video fragmentation and reverse image search**[25] that has been developed by CERTH in the past (see Fig. 4.1.3). This tool allows the user to submit a video for analysis (either by providing its URL or via uploading a local copy of it from his/her machine) and extract a set of representative keyframes, using the method described in (Teyssou et al., 2017). This method decomposes a video into sub-shots by assessing the visual resemblance of neighboring video frames with the help of the Discrete Cosine Transform (DCT). After analyzing a set of sampled frames, it produces a series of similarity scores and identifies sub-shot boundaries by spotting changes in the similarity tendency. Then, the frames that correspond to the point in time where the change of visual content is most pronounced are chosen as keyframes. The user can either monitor the progress of the analysis on the User Interface (UI) of the tool or close the browser and be notified by e-mail when the analysis results are ready (in the optional case that s/he provided an email address). The provided set of representative keyframes after the end of the analysis (see the left part of Fig. 4.1.4), allows the user to get an overview of the video content and perform keyframe-based reverse search for the video on the Web. For this, the user must right click on a keyframe and select one of the different supported search engines that are listed in the appeared pop-up menu (see the right part of Fig. 4.1.4). This process can be repeated as many times as needed using different keyframes, to assist the detection of near-duplicates of the video on the Web. The results of each search are presented in a new tab of the browser, which opens automatically (see Fig. 4.1.5).

[25] Accessible at: https://multimedia3.iti.gr/video_fragmentation/service/start.html.


Information Technologies Institute
Multimedia Knowledge and Social Media Analytics Laboratory
VIDEO ANALYSIS 4ALL
Submit
About this tool

On-line service for video fragmentation and reverse image search

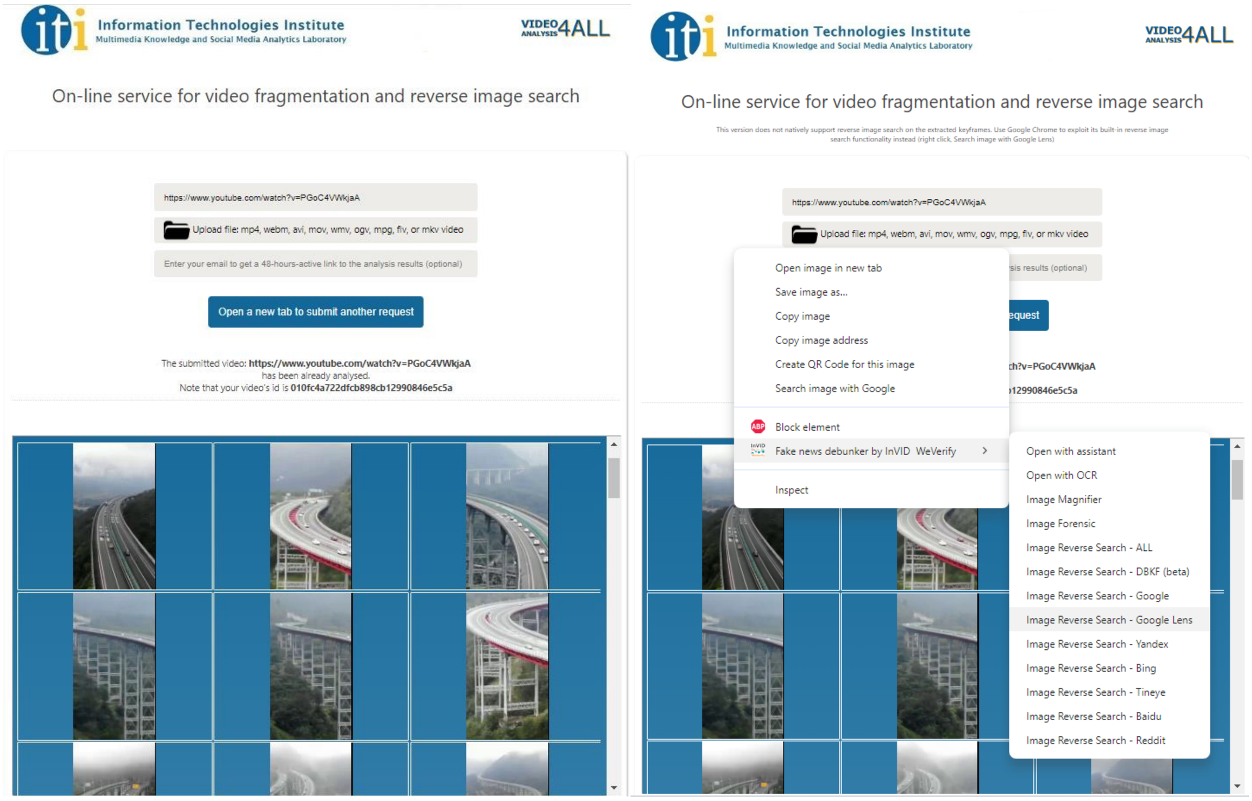
Insert the video URL (for supported video platforms please click About this tool)

Upload file: mp4, webm, avi, mov, wmv, ogv, mpg, flv, or mkv video

Enter your email to get a 48-hours-active link to the analysis results (optional)

© CERTH-ITI, MKLab

Fig. 4.1.3. The User Interface of the web-based tool for video fragmentation and reverse image search, that was used as the basis for our developments in AI4TRUST.


Information Technologies Institute
Multimedia Knowledge and Social Media Analytics Laboratory
VIDEO ANALYSIS 4ALL
Open a new tab to submit another request
Open image in new tab
Save image as...
Copy image
Copy image address
Create QR Code for this image
Search image with Google
Block element
Fake news debunker by InVID WeVerify
Inspect
Open with assistant
Open with OCR
Image Magnifier
Image Forensic
Image Reverse Search - ALL
Image Reverse Search - DBKF (beta)
Image Reverse Search - Google
Image Reverse Search - Google Lens
Image Reverse Search - Yandex
Image Reverse Search - Bing
Image Reverse Search - Tineye
Image Reverse Search - Baidu
Image Reverse Search - Reddit

On-line service for video fragmentation and reverse image search

https://www.youtube.com/watch?v=PGoC4VWkjaA

Upload file: mp4, webm, avi, mov, wmv, ogv, mpg, flv, or mkv video

Enter your email to get a 48-hours-active link to the analysis results (optional)

The submitted video: **https://www.youtube.com/watch?v=PGoC4VWkjaA**
has been already analysed.
Note that your video's id is **010fc4a722dfcb898cb12990846e5c5a**

On-line service for video fragmentation and reverse image search

This version does not natively support reverse image search on the extracted keyframes. Use Google Chrome to exploit its built-in reverse image search functionality instead (right click, Search image with Google Lens)

https://www.youtube.com/watch?v=PGoC4VWkjaA

Upload file: mp4, webm, avi, mov, wmv, ogv, mpg, flv, or mkv video

Fig. 4.1.4. Left side: the set of extracted keyframes from the submitted video for analysis. Right side: the pop-up menu that appears after right clicking on a keyframe and allows to perform reverse search through various search engines.

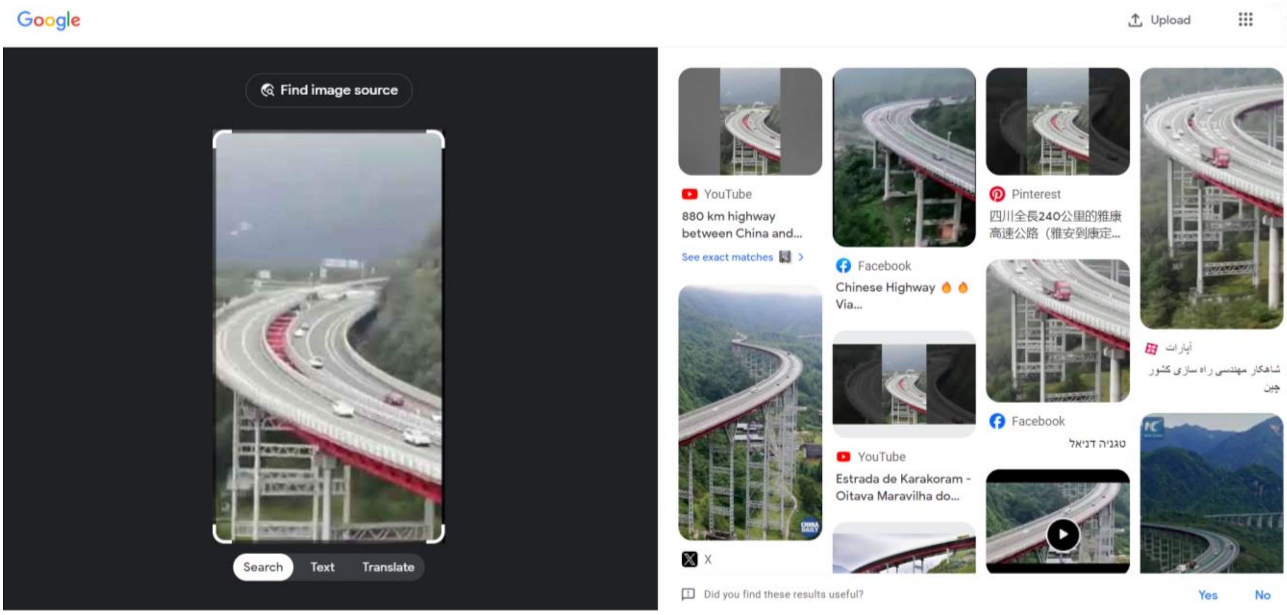

Google
Upload
Find image source
Search
Text
Translate
YouTube
880 km highway between China and...
See exact matches
Facebook
Chinese Highway 🔥🔥 Via...
Pinterest
四川全長240公里的雅康高速公路（雅安到康定...
آپارات
شاهکار مهندسی راه سازی کشور چین
Facebook
טגניה דניאל
YouTube
Estrada de Karakoram - Oitava Maravilha do...
X
Did you find these results useful?
Yes
No

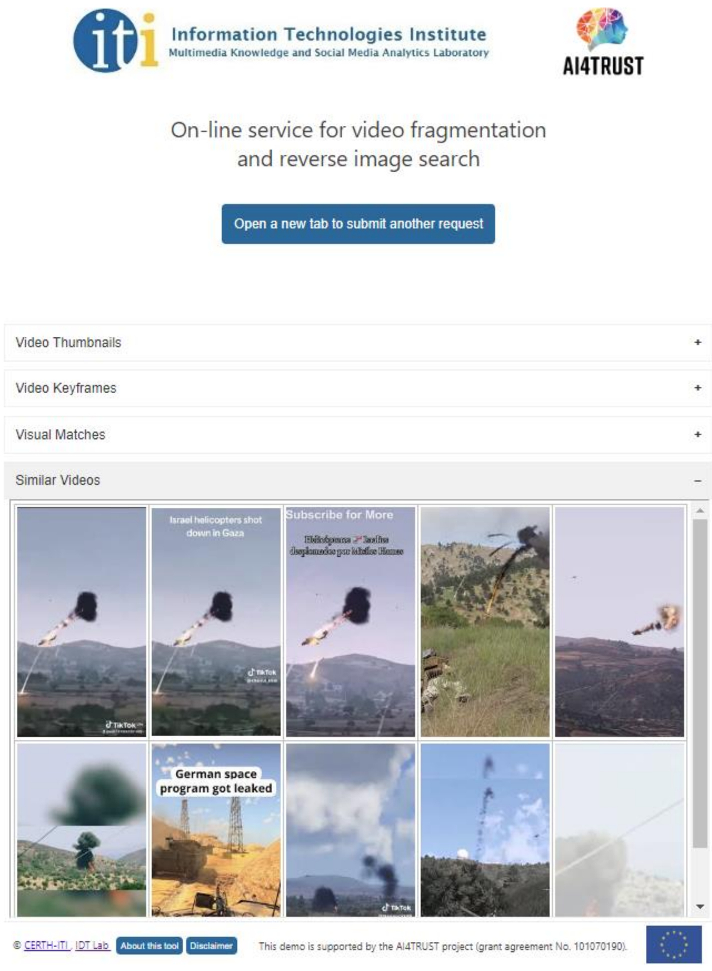
Fig. 4.1.5. The obtained results after performing reverse search using the keyframe on the left and the GoogleLens technology.

To **further facilitate the detection of near-duplicates**, we extended the backend of the aforementioned web-based tool by: i**) integrating a state-of-the-art AI-based method for video thumbnail selection** (Apostolidis et al., 2023), and ii) **automating the interaction between the tool and the search engine.** The employed video thumbnail selection method picks a number of frames (10 in total) by taking into account estimates about their aesthetic quality, representativeness and visual diversity, and using a frame picking mechanism which demotes the selection of frames that are visually-similar to the already picked ones. The selected set of frames (called thumbnails in the following) is used to search for near-duplicates of the video on the Web, using a search engine. To automate the searching process, we initially experimented with the Cloud Vision API of Google[26] and the integrated technology for detecting Web references to a given image[27]. However, the obtained results for a set of images indicated that the aforementioned technology performs poorly, as it systematically fails to detect references that are successfully identified by the GoogleLens technology. So, we decided to use GoogleLens for performing the thumbnail-based search. Given the fact that there is no publicly-accessible API for GoogleLens, we utilized a third-party API[28] that allows users to submit an image to GoogleLens and performs real-time scraping of the visual search results. Among other, the returned (JSON-structured) results from the utilized third-party API after an image-based search contain information about the most similar videos (i.e., video title, URL, source, channel, length, and thumbnail) and visual matches (i.e., webpage name and URL,

[26] https://cloud.google.com/vision
[27] https://cloud.google.com/vision/docs/detecting-web
[28] https://www.searchapi.io/

image URL, height and width) on the Web. Based on the above, we updated the backend of the web-based tool, in order to: i) **establish communication with the aforementioned third-party API**; ii) **submit each one of the selected video thumbnails to GoogleLens**; and iii) **retrieve the searching results.**

To allow internal testing and speed-up developments, we created **a custom UI** - independently from the foreseen UI for this technology in the AI4TRUST platform - that shows the: i) **set of selected video thumbnails** that were used for the automated search; ii) **set of extracted video keyframes** that can be used for further investigation following the previous, interactive approach; iii) **retrieved online sources** with visual matches of the selected thumbnails; and iv) **retrieved similar videos** from the Web. To minimize the user's burden to navigate through all these different results, the UI presents them under different collapsed regions, and shows, by default, **only the list of the similar videos** (see Fig. 4.1.6). If need be, this UI can be used to **assist the evaluation of the updated technology for reverse video search on the Web**, during the first pilot testing of AI4TRUST.



On-line service for video fragmentation and reverse image search

This demo is supported by the AI4TRUST project (grant agreement No. 101070190).

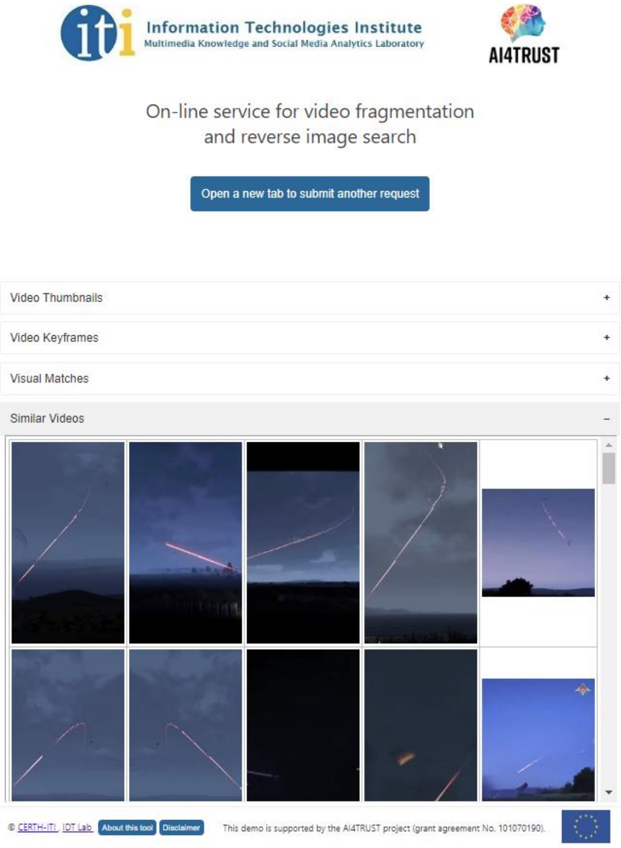
Information Technologies Institute
Multimedia Knowledge and Social Media Analytics Laboratory
AI4TRUST
Open a new tab to submit another request
Video Thumbnails +
Video Keyframes +
Visual Matches +
Similar Videos –
Israel helicopters shot down in Gaza
Subscribe for More
German space program got leaked
© CERTH-ITI , IDT Lab
About this tool
Disclaimer

Fig. 4.1.6. The analysis results for a query video, in the User Interface of the updated web-based tool for reverse video search on the Web.

4.1.4. Results and outlook

To assess the extent to which the **extended version of the tool facilitates the debunking of fakes that rely on video re-use**, we tested in a set of cases that have been reported on the Web[29]. In the following, we demonstrate the effectiveness of this new version through **two indicative examples**.

A 5-min. and 11-sec. video was posted on Facebook on May 16, 2021[30] alongside a claim that it shows a conflict between Israel and Hamas. It was shared with a similar claim on Facebook[31,32] and aired on the local TV channel Siyatha news[33]. Nevertheless, the footage was taken from a military-themed video game, called ARMA 3. After analyzing the video from the initial post with our tool, we get a set of similar videos, as depicted in Fig. 4.1.7. After clicking on the first two of them (top-ranked according to their similarity with the video under investigation), we are directed to TikTok videos showing short segments from the ARMA 3 video game (see Fig. 4.1.8), thus quickly finding a source for debunking the fake.

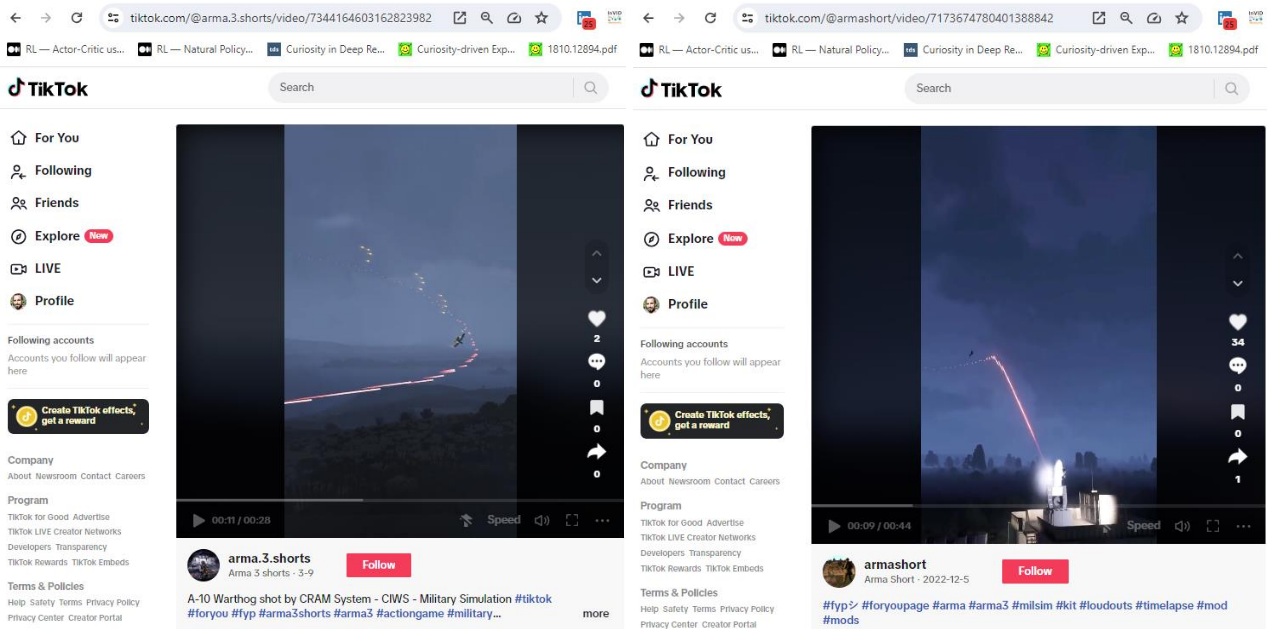

Information Technologies Institute
Multimedia Knowledge and Social Media Analytics Laboratory
AI4TRUST
Open a new tab to submit another request
Video Thumbnails +
Video Keyframes +
Visual Matches +
Similar Videos –
© CERTH-ITI, IDT Lab. About this tool Disclaimer

On-line service for video fragmentation and reverse image search

This demo is supported by the AI4TRUST project (grant agreement No. 101070190).

Fig. 4.1.7. Retrieved similar videos for the video under investigation.

[29] https://news.google.com/search?q=%22fact%20check%22%20%22keyframes%22&hl=en-US&gl=US&ceid=US%3Aen
[30] https://www.facebook.com/konappuwaa007/posts/835615127371123/
[31] https://www.facebook.com/Kamburupitiya.lk/videos/1610574579332824
[32] https://www.facebook.com/cyril.baddegama/posts/4038116546237697/
[33] https://www.facebook.com/groups/1903437133085969/permalink/3939888449440817



tiktok.com/@arma.3.shorts/video/7344164603162823982

arma.3.shorts
Arma 3 shorts · 3-9

A-10 Warthog shot by CRAM System - CIWS - Military Simulation #tiktok #foryou #fyp #arma3shorts #arma3 #actiongame #military...

tiktok.com/@armashort/video/7173674780401388842

armashort
Arma Short · 2022-12-5

#fypシ #foryoupage #arma #arma3 #milsim #kit #loudouts #timelapse #mod #mods

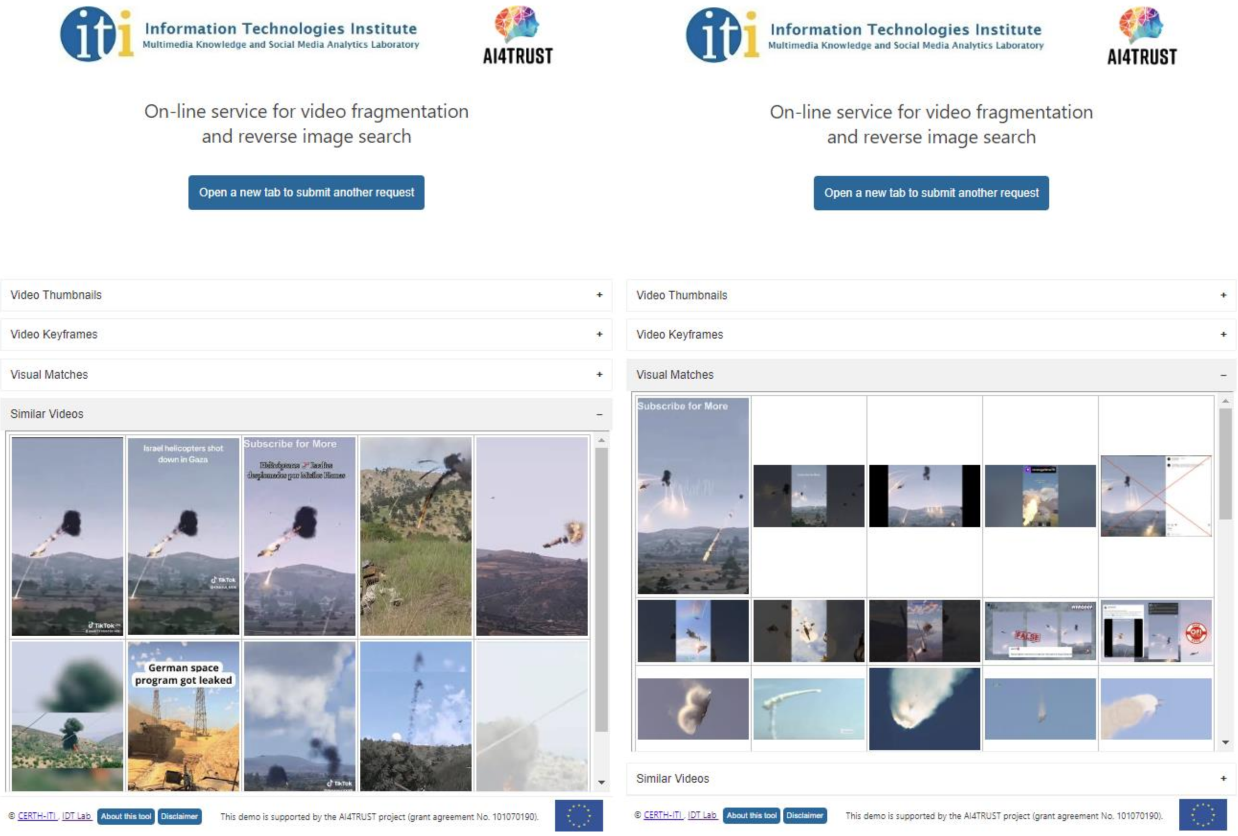
TikTok
Search
For You
Following
Friends
Explore New
LIVE
Profile
Following accounts
Accounts you follow will appear here
Create TikTok effects, get a reward
Company
About Newsroom Contact Careers
Program
TikTok for Good Advertise
TikTok LIVE Creator Networks
Developers Transparency
TikTok Rewards TikTok Embeds
Terms & Policies
Help Safety Terms Privacy Policy
Privacy Center Creator Portal
00:11 / 00:28
Speed
2
0
0
0
Follow
more
00:09 / 00:44
34
0
0
1

Fig. 4.1.8. Two TikTok videos showing short segments from the ARMA 3 video game.

Another 22-sec. video was posted on X (formerly Twitter) on October 7, 2023[34] (so, right after the attack of HAMAS in Israel) with the claim that Palestinian freedom fighters shot down four Israeli war helicopters in Gaza. The footage showed rockets bombing helicopters. However, the original video was posted on YouTube by @SeveralSim[35], a realistic graphics gaming channel. So, after analyzing the video with our tool, we get a set of videos (see the left side of Fig. 4.1.9) that are highly similar with the submitted but cannot help to assess its veracity. Though, looking at the retrieved visual matches (see the right side of Fig. 4.1.9), we see three different online sources that examine the veracity of the claim in the post under investigation. After visiting them (see Fig. 4.1.10), we see that our tool provided access to fact-checking articles that debunk this claim.

[34] https://twitter.com/Cricbaazharry/status/1710714336547917964?t=rctqaxoFo_1nFtlaPSTTCw&s=08

[35] https://www.youtube.com/@SeveralSim



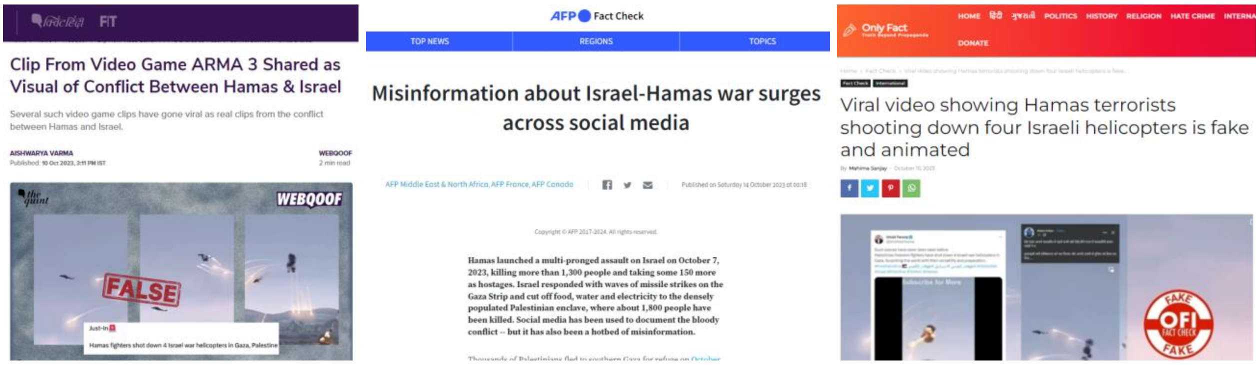
Information Technologies Institute
Multimedia Knowledge and Social Media Analytics Laboratory

On-line service for video fragmentation and reverse image search

© CERTH-ITI, IDT Lab

This demo is supported by the AI4TRUST project (grant agreement No. 101070190).

Fig. 4.1.9. Retrieved similar videos (left) and visual matches (right) for the video under investigation.



Clip From Video Game ARMA 3 Shared as Visual of Conflict Between Hamas & Israel

Several such video game clips have gone viral as real clips from the conflict between Hamas and Israel.

AISHWARYA VARMA
Published: 10 Oct 2023, 3:11 PM IST

WEBQOOF
2 min read

Misinformation about Israel-Hamas war surges across social media

AFP Middle East & North Africa, AFP France, AFP Canada

Published on Saturday 14 October 2023 at 00:18

Copyright © AFP 2017-2024. All rights reserved.

Hamas launched a multi-pronged assault on Israel on October 7, 2023, killing more than 1,300 people and taking some 150 more as hostages. Israel responded with waves of missile strikes on the Gaza Strip and cut off food, water and electricity to the densely populated Palestinian enclave, where about 1,800 people have been killed. Social media has been used to document the bloody conflict -- but it has also been a hotbed of misinformation.

Viral video showing Hamas terrorists shooting down four Israeli helicopters is fake and animated

Fig. 4.1.10. Retrieved online sources that debunk the claim that relates with the video under investigation.

The examples above show that the updated version of **the tool facilitates significantly the detection of near-duplicates of the submitted video on the Web**. Instead of requiring the user to manually select a keyframe to perform reverse image search through a search engine (as in the original version), the new tool looks for similar videos on the Web based on a set of automatically-selected video thumbnails. Thus, the investigation for near-duplicates is based on a broad set of representative video frames (thumbnails), while the burden for selecting these frames is removed from the user's shoulders. Moreover, the previously-needed amount of interaction between the user and the results of the analysis (video keyframes) and the search engine after performing a number of reverse image searches (retrieved online sources) is significantly reduced, as the submission of the selected thumbnails for reverse image search and the processing of the searching results is done in a fully-automated manner. This reduction was quantified by counting the number of clicks that a user has to do for debunking a set of fakes that rely on video re-use (10 in total), using the previous (baseline) and the updated version of the tool. This experimental comparison showed that a user of the previous version of the tool needs to select, on average, 4 keyframes for reverse image search and check, on average, 10 online sources from the retrieved ones (by GoogleLens). On the contrary, a user of the updated version of the tool does not need to manually select any keyframe and is able to find a near-duplicate of the query video or an online source debunking the claim after 2.5 clicks, on average.

Over the next months, we will further **extend this technology**, by: i) **automating interaction with additional search engines** (at least Bing, Yandex, Baidu); ii) **collecting and providing to the user information about the publication date of the retrieved videos** (an important factor for detecting re-use of a video from the past); and iii) **integrating a mechanism that will allow to look for near-duplicates of the video under investigation**, also in closed collections.

4.1.5. Exposed API for integration

The exposed **API** is **REST-based** and is built in Python. It contains three endpoints that are used for: i) submitting a video for analysis, ii) periodically checking the status of the analysis, and iii) retrieving the analysis results. The POST endpoint for submitting a video for analysis accepts a JSON object, like the following example:

```
Endpoint: http://multimedia2.iti.gr/video_analysis_v02/rvs
{
    "video_url": "the URL of the video",
    "user_key": "a unique 32-digits access key that allows access to the service",
    "Kf_num_sb": "an optional argument defining the number of extracted keyframes per
video fragment (default value: 3)"
}
```

After receiving an analysis request, this endpoint provides a JSON-structured reply, similar to the following one:

```
{
    "message": "The REST call has been received. Please check the status of the
analysis via the appropriate REST call",
    "video_id": "a unique 28-digits id that can be used to check the status of the
analysis and retrieve the analysis results"
}
```

As a note, for fetching videos from the Web, we use youtube-dl, a third-party component which supports fetching from an extensive list of online sources: https://github.com/ytdl-org/youtube-dl.

The GET endpoint for checking the status of the analysis can be called as shown below:

```
Endpoint: http://multimedia2.iti.gr/video_analysis_v02/status/video_id
```

After receiving a request, this endpoint provides a JSON-structured reply, as follows (the integer at the end of the status message indicates the progress of each step of the analysis as percentage):

```
{
    "status": "VIDEO_SUBSHOT_SEGMENTATION_STARTED:::57"
}
```

The GET endpoint for retrieving the analysis results can be called as in the following example:

```
Endpoint: http://multimedia2.iti.gr/video_analysis_v02/json/video_id
```

The JSON-structured response for the processed video is presented below:

```
{
    "generated_at": "2024-04-11 15:13:34.258506",
    "expires_at": "2024-04-13 15:13:34.258506",
    "generated_by": "https://multimedia2.iti.gr/video_analysis_v02",
    "version": "v12",
    "session": "8cfa9b4fad21e27714b7277c28e1fe929a69bc1c",
    "framerate": "24.000",
    "similar_videos": [
        {
            "url": "https://www.youtube.com/watch?v=36G-hQV3Wyg",
            "thumbnail": "https://encrypted-tbn1.gstatic.com/images?q=tbn:ANd9GcRxtZi
SFzjWtQTS7Tfu-yD0wKPbjo98hsTAcqXeyzC54CuK1aIE"
        },
             ...
    ],
    "visual_matches": [
        {
            "url": "https://www.facebook.com/CivilEngDis/videos/chinese-highway-via-
civil engineeringdiscoveries/348844739127560/",
            "thumbnail": "https://encrypted-tbn2.gstatic.com/images?q=tbn:ANd9GcScYh
oyu_cfclz1eKgSsoWhRsVQG57KXO3UXgSc0myWk-4w49xf"
        },
             ...
    ],
    "thumbnails": [
        {
            "time": "15.000",
            "url": "https://multimedia2.iti.gr/video_analysis_v02/thumbnail/8cfa9b4fa
d21e27714b7277c28e1fe929a69bc1c/thumbnail_360"
        },
             ...
    ],
    "subshots": [
        {
            "begintime": "0.042",
            "endtime": "1.250",
            "keyframes": [
                {
                    "time": "0.333",
                    "url": "https://multimedia2.iti.gr/video_analysis_v02/keyframe/
8cfa9b4fad21e27714b7277c28e1fe929a69bc1c/subshot0_1"
                },
                {
                    "time": "0.667",
                    "url": "https://multimedia2.iti.gr/video_analysis_v02/keyframe/
8cfa9b4fad21e27714b7277c28e1fe929a69bc1c/subshot0_2"
                },
                {
                    "time": "1.000",
                    "url": "https://multimedia2.iti.gr/video_analysis_v02/keyframe/
8cfa9b4fad21e27714b7277c28e1fe929a69bc1c/subshot0_3"
                }
            ]
        },
             ...
    ],
    "keyframes_zip": "https://multimedia2.iti.gr/video_analysis_v02/keyframes/8cfa9b4
fad21e27714b7277c28e1fe929a69bc1c",
    "thumbnails_zip": "https://multimedia2.iti.gr/video_analysis_v02/thumbnails/8cfa9
b4fad21e27714b7277c28e1fe929a69bc1c"
}
```

Besides some general information about the processing session (e.g., the generation and expiration time of the analysis results), this response contains: i) a list of links to the detected similar videos by the utilized search engine, and a list of links to the publicly-available thumbnails for these videos, which can be used for visualization purposes in the corresponding user interface of the AI4TRUST platform; ii) a list of links to online sources that contain visual matches to the thumbnails that were used in the automated search for near-duplicates of the video on the Web, and a list of links to the matched thumbnails/images; iii) a list of links to the selected video thumbnails (by the integrated thumbnail selection method) that were used for the automated search, and the appearance time of these thumbnails in the video; iv) the list of detected video fragments (by the integrated video subshot segmentation method) where each fragment is associated with information about the starting and ending time, and the link and timestamp of the three extracted keyframes that can be used for further investigation; and v) two links to zipped files containing the extracted video keyframes and thumbnails.

4.2. Deepfake image/video detection

4.2.1. Problem statement

The landscape of fake digital media, and specifically the generation and detection methods as well as their implications, have been described in detail in deliverable D2.1 (see Section 1.3.2). Recently, the progress of generative AI, and more precisely the visual content generation domain, has given rise to a vast amount of **publicly available methods and tools which can be used without requiring technical skills in order to create fake images and videos, the so-called deepfakes** (Tolosana et al. 2020). Such tools can create fully synthetic faces of persons that do not exist[36], swap faces in videos[37], reenact faces[38], or manipulate facial attributes such as hair, age, and eyeglasses[39]. The consequences of uncontrolled spread of such software can be **extremely harmful** for individuals as well as the society itself, as the potential malicious uses of this kind of technology range from fake pornography to hoaxes, identity theft, and even financial fraud. To make matters worse, the quality of deepfakes is becoming better and better and the generated content more and more realistic, rendering the detection of manipulation almost impossible for humans. Hence, the development of **automatic solutions that reliably detect deepfakes** is of utmost importance and is the topic of research that we conducted and present in this section.

4.2.2. Related work

Over the course of the project, our work on deepfake image/video detection was based on the **state-of-the-art method of Kumar et al. (2020)**. This method uses a triplet loss for deep metric learning and a deep neural classifier for video frame binary classification in pristine vs. falsified classes. Due to face pose variation, Kumar et al. (2020) also consider a combination of recurrent (LSTM/GRU) layers and 3D Convolutional layers for the extraction of information across the temporal and spatial domains. The method's predictions are computed by a mean voting operation of rolling frame predictions and is evaluated in highly compressed low-quality videos. Finally, this method exhibits increased performance on manipulated video classification in the CelebDF v2 (Li et al. 2020) and the FaceForensics++ (Rossler et al. 2019) datasets. Based on these observations, we also consider a pipeline involving triplet loss-based model learning due to its **increased representational efficiency on image datasets compared to features learned from last layers of classic CNNs**, such as the ones extracted using FaceNet (Schroff et al. 2015).

[36] https://thispersondoesnotexist.com/
[37] https://github.com/deepfakes/faceswap
[38] https://arxiv.org/abs/2007.14808
[39] https://arxiv.org/abs/1711.09020

4.2.3. Proposed method

Network architecture

The baseline model architecture we considered is the **InceptionResnetV3**[40] pre-trained on the face recognition task using the **VGGFace2 dataset** (Cao et al. 2018), as described in Schroff et al. 2015. In addition, we also considered **EfficientNet, XceptionNet** and the much larger **DeiT** (Touvron et al. 2021) and **ViT** (Dosovitskiy et al. 2020) **Transformer models pre-trained on ImageNet** as feature extractors. The **training** was conducted with **(anchor, positive, negative) triplets** having the objective to bring closer the embeddings of the anchor and the positive faces, while increasing the distance between the embeddings of the anchor and the negative faces[41]. We also considered **online triplet mining**, with easy and semi-hard mining strategies for positive and negative triplet pairs respectively, which results in the most informative training signal as suggested in Xuan et al. 2020. Finally, for the frame-level evaluation the extracted embeddings were used to **train an ML model** (i.e., Random Forest and MLP are considered) on the forgery task to assess their effectiveness. For the video-level evaluation an **LSTM network** was first applied on top of the extracted features to incorporate the temporal dependencies.

Used data

To establish a working pipeline, we considered a **small-scale dataset**, namely the c23 compression version of the FaceForensics++ dataset (Rossler et al. 2019). We sampled 25 frames per video, resulting in 24K frames per class (real/fake). For testing, we evaluated the models in 1,000 frames from 40 videos that were kept unseen during the model training. Additionally, for **evaluating our approach** we took into account the following standard deepfake detection benchmarks: **Deepfake Detection Challenge (DFDC)** (Dolhansky et al. 2020), **WildDeepFake** (Zi et al. 2020), and **CelebDF v2** (Li et al. 2020). These datasets resemble real-world cases, thus providing us with an **estimation of the in-the-wild performance of our method**.

Pre-processing

Prior to model training, the facial regions were detected with MTCNN[42] (by setting the detected face bounding box margin to 1.5) **and cropped**, as the deepfake manipulation is conducted on the face regions with surrounding background region information. This aims to include inconsistencies and artifacts introduced in the deepfake image during generation, as suggested in Rossler et al. 2019, Charitidis et al. 2020. Also, the pre-processing introduced in (Charitidis, et. al. 2020) was incorporated in the pipeline. This pre-processing involves **filtering false positive detections of MTCNN face detections** by applying an empirical threshold to the similarities of embeddings of the deepfake frames. The pre-processing stage yields video-level face clusters of increasing

[40] Implementation taken from https://github.com/davidsandberg/facenet

[41] This is implemented based on the TripletMarginLoss https://kevinmusgrave.github.io/pytorch-metric-learning/losses/#tripletmarginloss

[42] https://github.com/ipazc/mtcnn

confidence importance which generate the final predictions for each video. Concerning the **frame-based models pretrained on Imagenet-1k**, the images were normalized channel-wise using the ImageNet mean and standard deviation, followed by standard image augmentations, e.g., the ones considered in (Baxevanakis et al. 2022), which prevent overfitting and result in more robust classifiers to real-world data.

Implementation details

In Tables 4.2.1 & 4.2.2, we illustrate the **implementation details for the CNN-based and the Transformer-based feature extractors**, respectively. Due to class imbalance (4:1 for manipulated:pristine) we adopted a sampling scheme, with class weights $W_i = \frac{N}{N_i}$ (N is the total dataset size, while N_i the size of class i), to counteract the skewed class distribution. Furthermore, it was ensured that no face appeared both in training and validation sets to mitigate overfitting and improve the generalization ability of the model. Following previous investigations of optimal Transformer model learning rate schedulers, cosine annealing warmup was utilized, which corresponds to linear increase of learning rate for 1,000 warm-up steps, followed by cosine annealing decay schedule. Finally, the LSTM network has 6 layers with hidden layer size 100.

Optimizer	Adam
Learning Rate **Exponential Decay**	0.0005 (gamm=0.9) yes
Epochs	90
Regularisation	Weight Decay: 0.05
	Dropout: 0.2
	Early Stopping: 10
Additional	Gradient Accumulation (accum. param. : 10)
	Stochastic Depth Probability: 0.2
Augmentations	Random Albumentations

Table 4.2.1: Training details for CNN models.

Optimizer	AdamW
Learning Rate (Multiplier): **Warmup Epochs:**	0.0003135 (DeiT) 0.003 (Vit) 5 (DeiT) 3.4 (ViT)
Epochs	90
Regularisation	Weight Decay: 0.05
	Dropout (MLP Head): 0.2
	Early Stopping: 10
Additional	Gradient Accumulation (accum. param. : 10)
	Stochastic Depth Probability: 0.1
Augmentations	RandAugment (p=0.5)
	Mixup (p=0.8)
	Cutmix (p=1)
	RandErasing(p=0.25)

Table 4.2.2: Training details for Transformer models.

- **Evaluation protocol**

As validation accuracy we defined the fraction of validation image triplets wherein the feature distance between the anchor and the positive image is less than the feature distance between the anchor and the negative. The **performance metrics** for evaluation on the test set we considered are **AUC and balanced accuracy**.

4.2.4. Results and outlook

Table 4.2.3 presents the performance of our pipeline on **4 datasets**, using different backbones and a Random Forest classifier on top of the extracted features. It is observed that the ImageNet pre-trained models exhibit better performance compared to the In.ResNetV1 (when trained only on one dataset). Also, the Transformer-based model outperforms the CNN-based ones in 1 out of 4 datasets. However, the In.ResNetV1 when trained on the combination of all datasets (FF++, DFDC, WildDeepFake, CelebDF v2) exhibits a consistently good performance outperforming the rest of the models on WildDeepFake and CelebDF v2, thus we deployed this model (details in Section 4.2.5).

	In.ResNetV1	**In.ResNetV1 (combined datasets training)**	**XceptionNet**	**EffiNetB0**	**EffiNetB7**	**ViT**
FF++ (HQ)	**0.998** / **0.998**	0.648 / 0.714	0.957 / 0.957	0.990 / 0.990	0.990 / 0.990	0.995 / 0.995
FF++ (LQ)	0.643 / 0.643	0.838 / 0.911	0.910 / 0.910	0.931 / 0.931	0.918 / 0.918	**0.943** / **0.943**
DFDC	0.667 / 0.667	0.824 / 0.910	**0.940** / 0.750	0.827 / 0.827	0.934 / **0.934**	0.853 / 0.853
WildDeepFake	0.600 / 0.640	**0.789** / **0.864**	0.728 / 0.711	0.649 / 0.651	0.650 / 0.661	0.684 / 0.658
CelebDF v2	0.500 / 0.481	**0.922** / **0.987**	0.500 / 0.492	0.905 / 0.905	0.839 / 0.839	0.500 / 0.479

Table 4.2.3. Performance of the proposed triplet network with different backbones and a Random Forest classifier on top, in terms of balanced accuracy and AUC, on FF++ (HQ & LQ), DFDC, WildDeepFake, and CelebDF v2 datasets. InceptionResNetV1 is pre-trained on VGGFace2, while XceptionNet, EfficientNet and Vit are pre-trained on ImageNet.

Table 4.2.4 compares pre-processing pipelines on 3 deepfake detection datasets. As observed, combining face detection with face alignment is the best pre-processing practice compared to using only one of the two.

	Preprocessing A	**Preprocessing B**	**Preprocessing C**
DFDC	0.601 / 0.610	0.565 / 0.560	0.496 / 0.470
CelebDF v2	0.572 / 0.582	0.544 / 0.551	0.478 / 0.481
FF++	0.514 / 0.525	0.498 / 0.503	0.511 / 0.536

Table 4.2.4. The impact of the preprocessing pipeline on test set performance (balanced accuracy / AUC) on the DFDC, CelebDF v2, and FF++ datasets. A: Face detect and alignment, B: Face detect only, C: Face alignment only.

Table 4.2.5 illustrates **how performance can be increased by considering larger training datasets.** More precisely, the results indicate that training on the combination of DFDC and CelebDF v2 significantly increases the performance in terms of balanced accuracy and AUC.

	DeiT + MLP (DFDC+CelebDF v2)	Deit +MLP (DFDC)	DeiT + MLP (CelebDF v2)
Balanced accuracy	0.610	0.559	0.470
AUC	0.609	0.472	0.474

Table 4.2.5. Performance of the proposed pipeline with DeiT as backbone and MLP as classifier on unseen deepfake videos, using different training sets.

In order to further improve performance, different candidate models were assessed collectively. The objective was to construct an optimal ensemble architecture of different models, which act supplementarily. First, we computed the Jaccard coefficient metric on individual experimental architectures, which is reported in Table 4.2.6. It is hypothesized that the Jaccard index of pairs of models, computed on their misclassified outcomes (i.e. Fp, Fn), could guide model selection for ensembling and such a model combination would improve the generalization capacity of the architectures. The corresponding results are presented in Table 4.2.7.

Model id	Backbone + head	Training set	{Tp,Fp,Tn,Fn}	Jaccard index
A	DeiT + MLP	DFDC	{11482,1916,1157,3985}	{0.847,0.505,0.642}
B	DeiT + MLP	CelebDF v2	{341,249,4827,812}	{0.152,0.155,0.24}
C	Xception + MLP	DFDC	{8497,4901,4271,871}	{0.494,0.844,0.630}
D	ViT + MLP	DFDC	{10827,2571,4428,714}	{0.357, 0.755, 0.362}

Table 4.2.6. Optimal model selection assessment of ensemble architectures for Deepfake Detection Task.

Ensemble	Jaccard index	Balanced accuracy	AUC
A+B	0.505	0.609	0.607
C+A	0.494	0.592	0.580
A+D	0.357	0.624	0.621

Table 4.2.7. Ensemble model results on test set. Model identification is shown in Table 6.

Given the challenging nature of the task and the availability of several options for different parts of the pipeline**, future work will perform a systematic evaluation of different options**, including more recent state-of-the-art methods on a **wide range of datasets and evaluation settings**, as well as **investigation of novel contributions on top of the best performing methodologies**.

4.2.5. Exposed API for integration

The exposed **API** serves responses for deepfake video detection, under the video/dml endpoint, based on **the In.ResNetV1 model trained on 4 datasets** (FF++, DFDC, WildDeepFake, CelebDF v2). All popular formats are supported, however the user should be aware that model performance may be different in cases of different formats, compression rates, or resolution as already stated in Section 2.2 of the Deliverable D2.1. The response structure for a video sample is presented below.

```
  "deepfake_video_report": {
    "completed": true,
    "gpu": true,
    "message": "Process completed successfully.",
    "number_of_shots": 2,
    "prediction": 0.003423452377319336,
    "prediction_time": 6.661271085031331,
    "total_number_of_frames_for_inference": 16,
    "video_path": "https://artifacts.mever.iti.gr/s3/deepfake-
public/videos/beee14fb383f248a1b569cdac399fd0d/video.mp4",
    "results": [
      {
        "message": "Prediction completed for the shot.",
        "prediction_time": 4.964080687612295,
        "shot": 0,
        "shot_start": 0.0,
        "shot_end": 9.0,
        "number_of_faces": 1,
        "components_info": [
          {
            "component_no": 0,
            "enriched_bboxes": [
              {
                "frame_id": 25,
                "timestamp": 1.0,
                "bbox": {
                  "left": 117,
                  "top": 94,
                  "right": 289,
                  "bottom": 266
                },
                "prediction": 0.00441133975982666
              },
              ...
```

4.3. Occlusion-robust Deepfake detection

4.3.1. Problem statement

With the rapid progress of deepfake production technologies, less traces of forgery are now easily detectable, especially in hard examples of varying facial expressions, head pose, and facial landmarks. The vast majority of detection methods rely on **complete faces**; however, this type of analysis is **less powerful** in unusual circumstances. In response to this, **deepfake detection on partial face areas has recently gained attention**.

4.3.2. Related work

Existing approaches mostly utilize **deep learning models to extract biometric features** such as facial landmarks (e.g., eyes, eyebrows, mouth). Matern et al. (2019) detect a fake face by the light reflections and color inconsistencies, while Nirkin et al. (2021) employ contextual association to detect inconsistencies in facial regions. In recent years, with the development of the **multihead attention mechanism and visual transformer**, some methods integrate the attention mechanism to make the discriminator focus more on partially manipulated areas. The visual transformer divides the person's face into patches and estimates an attention map to assess forgery per block. Although some methods have focused on facial-region deepfake detection, most of them still detect forgeries region-by-region without considering landmark-level detection. As the forgery methods tend to diversify, the impersonated individuals may wear masks, sunglasses, or other accessories, which will cause part of the face to be occluded. The features extracted from the original methods are suboptimal in such cases, leading to poor detection performance. At the same time, to avoid being detected, some of the forged media are safeguarded with cropping, masking, or down-sampling, which will also reduce the accuracy of existing deepfake detection methods. To the best of our knowledge**, the only deepfake detection research addressing forgery detection with occluded facial landmarks is Xue et al. 2022**, which considers a combination of landmark-based and whole-face representation learning to address deepfake detection on occluded faces.

4.3.3. Proposed method

Network architecture

Experiments were conducted based on the architecture presented in Figure 4.3.1, which considers a **multi-branch approach for face-level and facial-landmark-level features extraction**. Some methodological modifications have been applied on the original model. More precisely, two Transformer-based approaches were explored, with **pre-trained Visual Transformers from Pytorch torchvision.models and custom Multi Head Attention blocks**:

- **Custom Visual Transformer.** The model consists of 7 distinct Transformer encoders, one for each facial landmark extracted from the images in the dataset (mouth, left eyebrow, right eyebrow, right eye, left eye, nose, jaw). Each transformer consists of 6 multi-head attention blocks with pre-LayerNormalization, which combine the learned query-key-value triplets of the patched images to 6 sub-queries, sub-keys, and sub-values to compute the attention maps. Each transformer is trained from scratch, learning to classify the organs as fake vs. original.

- **Pre-trained Visual Transformer**: For the Transformer encoder a variant of BERT architecture is utilized, with pretrained weights on ILSVRC-2012 ImageNet.

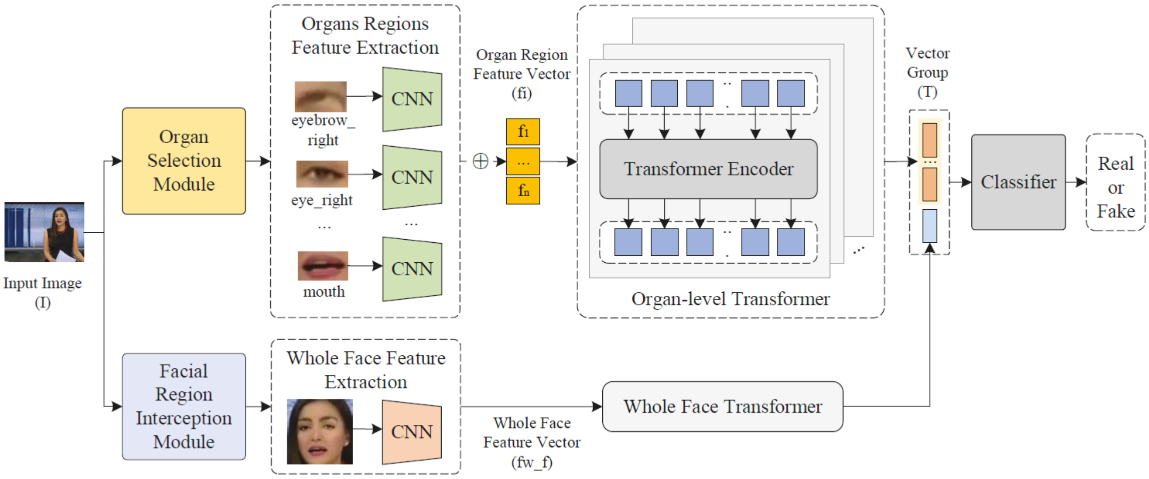


Figure 4.3.1. The model architecture we experimented with, adopted from (Xue et al 2022).

Used data

We considered the **FaceForensics++ dataset** which contains 44,335 images in total (fake/original). We used 35,468 of them for training and the remaining 8,867 for validation. Also, for evaluation we utilized the FO**FDTD dataset** which has 750 authentic images, 750 GAN-generated images, and 900 forgery images made by humans. Moreover, the depicted individuals' faces are partially occluded by masks and sunglasses.

Implementation details

dlib[43] was used to detect 68 facial landmarks and the corresponding bounding box around each landmark per image. We considered the **AdamW optimizer** (base learning rate is set to 0.0001, and weight decay to 0.1) for **the training of all organ Transformers and the face-level Transformer**. We increased the learning rate linearly for 1000 warmup steps and then decreased it proportionally to the inverse square root of the step number. We summed the binary cross entropy loss values from individual predictions, to update the Classifier's parameters (cf. Figure 4.3.1). Individual component's loss gradient was used to update the Transformers of each branch.

[43] https://github.com/davisking/dlib

Evaluation protocol

For evaluation, **the maximum vote from all landmark and face predictions** was determined for each batch and used as the model predicted output, while **the performance was measured based on balanced accuracy and AUC**.

4.3.4. Results and outlook

Improving training data quality

Undetected landmarks or landmarks of poor image quality (dlib detection failure) were discarded from the input and not used to update the classifier of the model. This is achieved by setting the image patch embeddings corresponding to the facial landmark that is occluded or of bad quality to zeros tensor of dimensionality 252. Therefore, providing a cleaner learning signal for the Vision Transformers and decreasing false detections. The “confidence” of face bounding box detection of the pretrained dlib HOG + Linear SVM regression model was used as measure of the quality of the detected face bounding box and a threshold - empirically determined - of 1.5 was set for minimum detection confidence for any of the 7 facial landmarks. The “confidence” value effectively removed faces detected partially and with missing landmarks (e.g., jawlines). Since the “confidence” of bounding box detection information was not sufficient to adequately filter individual landmark detections of bad quality, the variance of the Laplacian - 2nd spatial derivative of image pixels per image dimension - was introduced as a coarse blur measure of the extracted facial landmarks images. Empirical variance thresholds were then determined and set for each facial landmark, in addition to the bounding box regression “confidence” value for clean image data preparation for model training. **Examples of extracted facial landmarks**, with superimposed Laplacian information, are presented in Table 4.3.1.

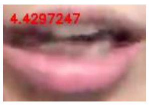
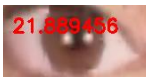
Landmark Description	“Mouth”	“Right Eye”	“Right Eyebrow”
Image, Variance of Laplacian			

Table 4.3.1. Example dlib facial landmark detections, along with the corresponding variance of the Laplacian.

Evaluation

The **performance of the Custom Visual Transformer architecture** was evaluated without the input of facial landmarks, to estimate baseline performance for comparison with the overall pipeline. The top row of Table 4.3.2 tabulates the validation metrics computed on 20% of FF++ data, from all fake image generation methods, when considering only the centered, cropped frame faces for training the Transformer and classifier pipeline. The remaining rows show **how the efficiency of the approach increases, when increasingly more facial organs are included in the detection pipeline.** We observe that utilizing only the face-level Transformer of the pipeline achieves a 78.7%

validation accuracy, while including the mouth, eyebrows, and the left and right eyes of the face boosts the performance to 89.5% accuracy.

Facial regions	Accuracy	AUC
Face	0.787	0.770
Face + mouth	0.811	0.791
Face + mouth + eyebrows	0.807	0.801
Face + mouth + eyebrows + eyes	0.895	0.880

Table 4.3.2. Performance in terms of accuracy and AUC for several input combinations of facial regions, on FF++ dataset.

The complete pipeline consisting of trained Visual Transformers for all facial organs, was evaluated in the FOFDTD dataset, which incorporates occlusions of facial landmarks by various objects (masks, sunglasses, etc.). The results presented in Table 4.3.3, show that in the case of GAN-generated occlusions, the performance is above random but still on low levels, while in the case of artificial occlusions the performance is near random.

Dataset	**Accuracy**	**AUC**
FOFDTD (GAN-Generated occlusions)	62.5%	62.5%
FOFDTD (Artificial occlusions)	53.0%	51.7%

Table 4.3.3. Performance on FOFDTD dataset.

4.3.5. Exposed API for integration

Not applicable as the functionality is the same as the one offered by the deepfake detection API described in 4.2.5, so in case of high detection accuracy, results will be directly integrated in the former API. However**, current results were not considered of sufficient accuracy to proceed with integration work**.

4.4. Sensational content detection

4.4.1. Problem statement

A recently published work (Hamby et al., 2024) pointed out that various guides to identifying disinformation anecdotally note **fake news' tendency to adopt a sensationalist story format** (Ireton & Posetti, 2018; PBS, 2021). This format might also include the use of **sensational visual content** that aims to attract the viewers' attention (as a first step toward the further spread of disinformation through social media). Based on the above observation, we argue that the **development of methods for detecting sensational visual content** could assist the identification of check-worthy media items and facilitate the work of fact-checkers. Moreover, similarly with text (Damstra et al., 2021; Gupta et al., 2022) the viewers' attention is more likely to be attracted using visual content that causes a negative sentiment. Building on this remark and taking into account the existing datasets in the relevant literature, our initial goal is to train a model for classifying visual content (images/videos) according to the presence (or not) of visually-disturbing content. Over the course of the project, we will extend our developments by taking into account other types of sensational content that is typically found in disinformation items and campaigns.

4.4.2. Related work

The literature in the broad category of sensational content detection primarily encompasses works on **NSFW detection**, including traditional shallow methodologies, such as Ap-Apid, et al., 2005; Lopes, et al., 2009; Santos, et al., 2012, as well as more recent deep learning-based methods, such as Moustafa, et al., 2015; Gangwar, et al., 2021; Fu, et al., 2021; Saxena, et al., 2023. Other works for **detecting harmful content**, i.e., content that evokes anxiety or fear, focus on violence detection, included handcrafted methods combined with non-visual modalities (Giannakopoulos, et al., 2006), as well as deep learning ones (Dai, et al. 2015; Mu, et al., 2016). A **challenging step** towards disturbing content detection is the **collection of data**, due to the nature of the images of the disturbing class, restricting, in turn, the generalization ability of the models trained with few images (Larocque, 2021). The **DID dataset** (Zampoglou, et al., 2016) can be considered as the largest one, considering the disturbing content detection task, being however a small dataset that consists of 5,401 images. To this end, a framework that exploits **large-scale multimedia datasets** to automatically extend initial training datasets with hard examples has been applied to the abovementioned dataset in Sarridis, et al., 2022.

4.4.3. Proposed method

We initially developed **a "version zero" set of state-of-the-art models for sensational content detection**, as a starting point for experimentation in the project. More specifically, **models of varying complexity** were trained for distinguishing between disturbing and non-disturbing images, including ResNet-18 (He, et al., 2016) , Wide-ResNet-50-2 (Zagoruyko, et al., 2016), EfficientNet-b0 (Koonce, et al., 2021), EfficientNet-b1, EfficientNet-b4, ViT-B-16 (Dosovitskiy, et al., 2020), SqueezeNet_1_1 (Iandola, et al., 2016), CLIP (ViT-L/14) (Radford, et al., 2021), and **a simple lightweight model**, providing a comprehensive comparative study. All the aforementioned models, apart from the latter one, are pre-trained on ImageNet weights, and adapted to address the binary classification task, by introducing a linear layer at the output with two neurons. Since the sensational content detection is a task of utmost importance, in order to enable the deployment on devices with limited computational resources, apart from the comparatively more lightweight models, such as the **SqueezeNet model**, we also trained a simple lightweight model. This model consists of **two convolutional layers** with six and sixteen kernels of size 5x5 respectively and three fully connected layers (128 × 64 ×2). A list of the trained network architectures along with the number of their parameters is provided in Table 4.4.1. All the models were trained using the **cross-entropy loss**, on the augmented dataset produced in Sarridis, et al., 2022. This dataset consists of 30,106 training images derived from the **DID** (Zampoglou, et al., 2016) and **YFCC** (Thomee, et al., 2016) **datasets**, and a set of 1,080 test images (DID dataset).

Furthermore, in the studied problem there are wide variations both in the **"disturbing" and "non-disturbing" classes.** For example, the former one may include images that contain violence, or animal cruelty, while in the latter one there is anything other than disturbing images. Thus, in order to exploit these variations for improving the classification performance, we worked towards exploring possible **sub-classes.** Since these subclasses are unknown, our approach is to develop auxiliary objectives aiming to reveal them during the training process. More specifically, considering the representations at the feature space generated by the penultimate layer of a model, we introduced an **additional auxiliary objective that forces the training samples** to approach their nearest representations, belonging to the same class.

Model	# Parameters
ResNet-18	11.1M
Wide-ResNet-50-2	66.8M
EfficientNet-b0	5.2M
EfficientNet-b1	7.7M
EfficientNet-b4	19M
ViT-B-16	85.8M
SqueezeNet_1_1	723K
CLIP (ViT-L/14)	150M
Lightweight	62K

Table 4.4.1. Trained models for sensational content detection along with the number of their parameters.

4.4.4. Results and outlook

The results of the comparative study are provided in Table 4.4.2. As it is demonstrated, **EfficientNet-b1 achieves the best performance**, a result that is consistent with the findings in (Sarridis, et al., 2022). Comparable performance is accomplished by the EfficientNet-b0 and the image encoder of the CLIP model. The SqueezeNet model attains remarkable performance, being also relatively lightweight. The lightweight model achieves the worst performance, which is reasonable, since it is an ultra-lightweight model trained from scratch. Regarding the applied approach for exploiting possible subclasses, the experimental results considering the EfficientNet-b1 and the lightweight model are presented in Table 4.4.3. These results indicate that **the automated reveal of subclasses of data through the additional objective, and the use of this knowledge during the training process is beneficial**, as it leads to improved performance in both of the considered methods.

In the future, we will explore the **use of Large Multimodal Model (LMM)** based knowledge in order to improve the detection performance. That is, LMMs such as **MiniGPT-4** (Zhu, et al., 2023) will be utilized to derive additional meaningful knowledge, in order to assist the model towards the sensational content detection task.

Model	Test Accuracy (%)
ResNet-18	92.130
Wide-ResNet-50-2	92.870
EfficientNet-b0	93.333
EfficientNet-b1	**93.426**
EfficientNet-b4	92.870
ViT-B-16	92.963
SqueezeNet_1_1	89.352
CLIP (ViT-L/14)	93.056
Lightweight	70.463

Table 4.4.2. Accuracy of the trained models for sensational content detection. Best performance in bold.

Method	Lightweight	EfficientNet-b1
Baseline	70.463	93.426
Subclass	**71.667**	**93.870**

Table 4.4.3. Accuracy for the proposed subclass criterion against the baseline.

4.4.5. Exposed API for integration

For the moment this technology is not exposed through an API. However, we plan to deploy the best-performing model in REST-based API that will be exposed to assist the integration of this technology in the AI4TRUST platform and the Disinformation Warning System.

4.5. Synthetic image/video generation

As indicated in the introduction, developing tools to **create visual data synthetically** is very relevant for the AI4TRUST project. These data can then be used to **further train and evaluate the developed deepfake (image/video) detection technologies.** The generated visual content is **complementary to the generated audio content** (described in Section 3.3.1).

4.5.1. PAIR Diffusion: A Multimodal Object-Level Image Editor

4.5.1.1. Problem statement

Generative image editing has recently witnessed extremely fast-paced growth. Some works use high-level conditioning such as text, while others use low-level conditioning. Nevertheless, most of them lack fine-grained control over the properties of the different objects present in the image, i.e., object-level image editing. To address this problem, we tackle the task by perceiving the images as an amalgamation of various objects and aim to **control the properties of each object in a fine-grained manner**. Out of these properties, we identify structure and appearance as the most intuitive to understand and useful for editing purposes. We propose **PAIR Diffusion**, a generic framework that can enable a diffusion model to control the structure and appearance properties of each object in the image. We show that having control over the properties of each object leads to comprehensive editing capabilities. Additionally, we propose a **multimodal classifier-free guidance** which enables editing images using both reference images and text when using our approach with foundational diffusion models.

4.5.1.2. Related work

When editing a real image, a user generally desires to have an intuitive and precise control over different elements (i.e., the objects) composing the image and to manipulate them independently. We can categorize the **existing image editing methods** based on the level of control they have over individual objects in an image. One line of work involves the use of text prompts to manipulate images (Brooks, et al., 2022; Hertz, et al., 2022; Liew, et al, 2022; Liu, et al, 2022). These methods have **limited capability for fine-grained control at the object level**, owing to the difficulty of describing the shape and appearance of multiple objects simultaneously with text. In the meantime, prompt engineering makes the manipulation task tedious and time-consuming. Another line of work uses low-level conditioning signals such as masks (Hu, et al., 2022; Patashnik, et al., 2023; Zeng, et al., 2022) sketches (Voynov, et al, 2022), images (Cao, et al, 2023; Song, et al, 2022; Yang, et al, 2023) to edit the images. However, most of these works either **fall into the prompt engineering pitfall or fail to independently manipulate multiple objects**.

4.5.1.3. Proposed method

In contrast to previous works, we aim to independently control the properties of multiple objects composing an image i.e., **object-level editing**. We show that we can formulate various image editing tasks under the object-level editing framework leading to **comprehensive editing capabilities**.

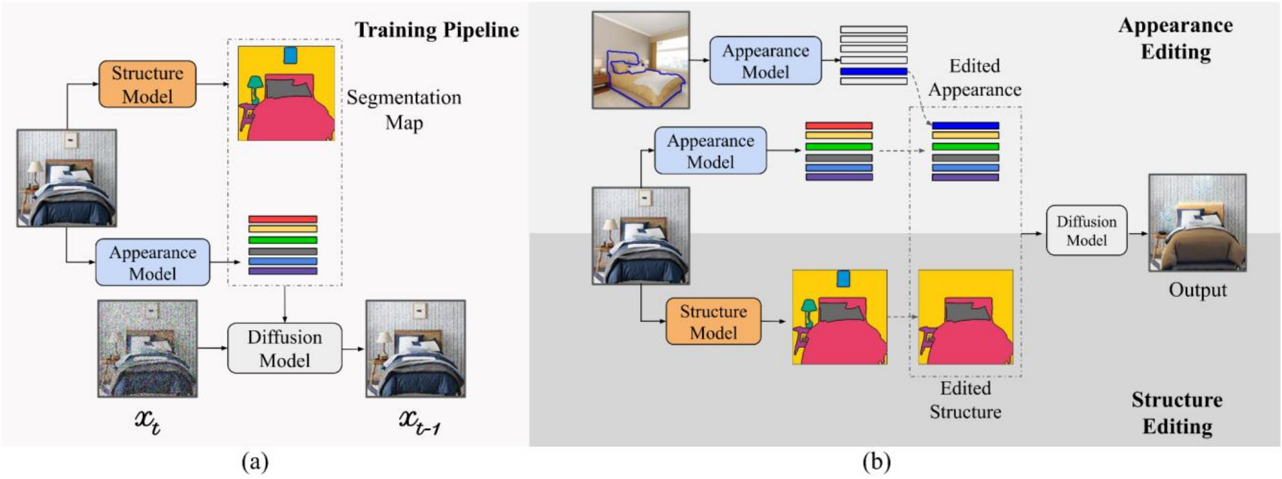


Figure 4.5.1.1. Overview of PAIR Diffusion. An image is seen as a composition of objects each defined by different properties like structure (shape and category), appearance, depth, etc. We focus on controlling structure and appearance. (a) During training, we extract structure and appearance information and conditionally train a diffusion model. (b) At inference, the framework supports multiple editing operations by independently controlling the structure and appearance of any real image at the object level.

To tackle the aforementioned task, we propose a novel framework, dubbed **Structure- and - Appearance Paired Diffusion Models (PAIR-Diffusion)** (see Fig. 4.5.1.1). Specifically, we perceive an image as an amalgamation of diverse objects, each described by various factors such as shape, category, texture, illumination, and depth. We then further identify **two crucial macro properties of an object: structure and appearance.** Structure oversees the object's shape and category, while appearance contains details like texture, color, and illumination. To accomplish this goal, PAIR-Diffusion adopts an off-the-shelf network to estimate panoptic segmentation maps as the structure, and then extract appearance representation using pre-trained image encoders. We use the extracted per-object appearance and structure information to condition a diffusion model and train it to generate images. In contrast to previous text-guided image editing works (Brooks, et al. 2022; Avrahami, et al, 2022; Couairon, et al, 2022; Ruiz, et al, 2022), we consider **an additional reference image to control the appearance.** Compared to text prompts which can only vaguely describe the appearance, images can precisely define the expected texture and make fine-grained image editing easier. Having the ability to control the structure and appearance of an image at the object level gives us comprehensive editing capabilities. Using our framework, we can **achieve localized free-form shape editing, appearance editing, editing shape and appearance simultaneously, adding objects in a controlled manner, and object-level image variation** (see Fig. 4.5.1.1).

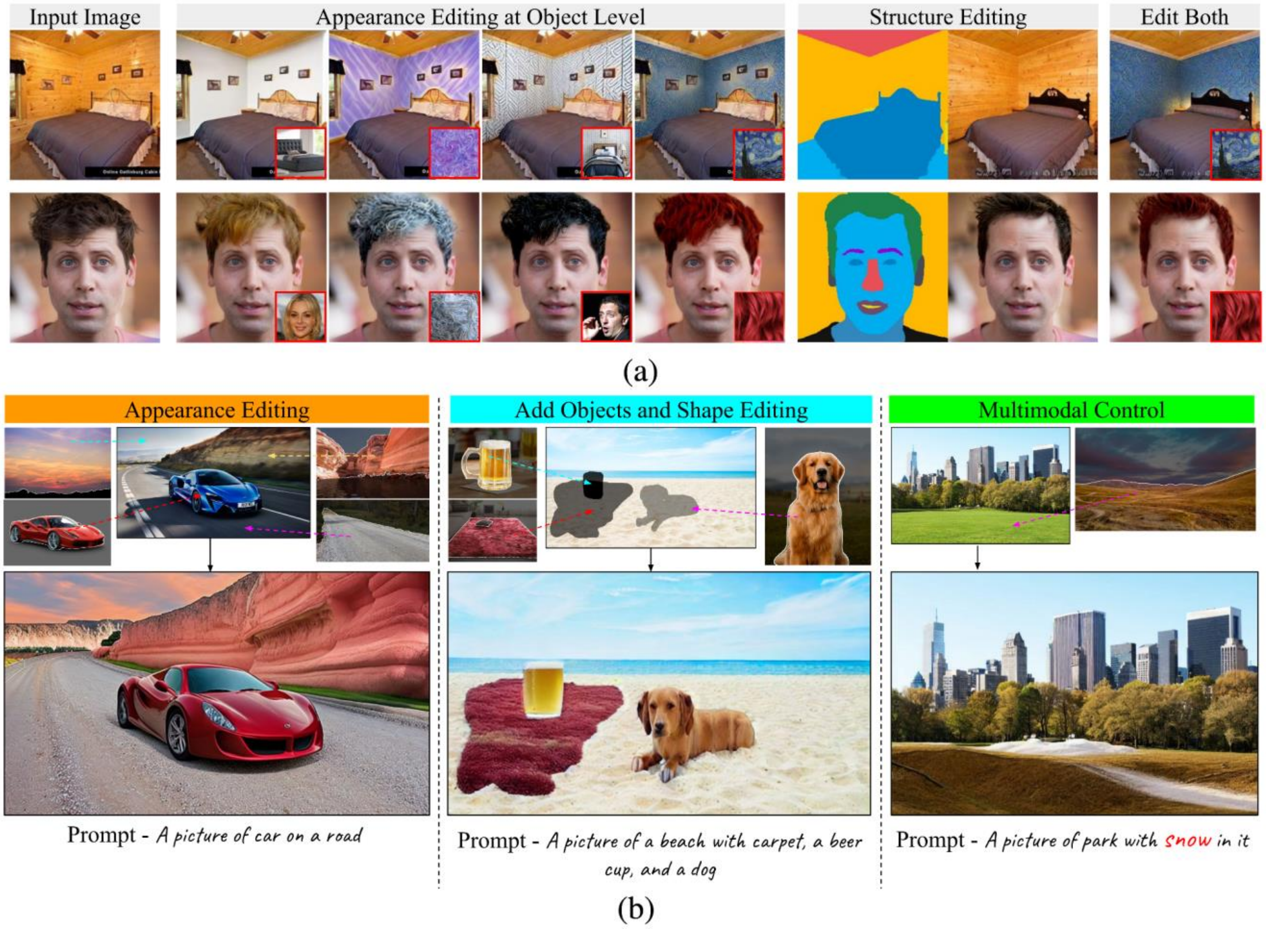

Input Image
Appearance Editing at Object Level
Structure Editing
Edit Both
(a)
Appearance Editing
Add Objects and Shape Editing
Multimodal Control
Prompt - A picture of car on a road
Prompt - A picture of a beach with carpet, a beer cup, and a dog
Prompt - A picture of park with snow in it
(b)

Figure 4.5.1.2. PAIR diffusion framework allows appearance and structure editing of an image at an object level. Our framework is general and can enable object-level editing capabilities in both (a) unconditional diffusion models and (b) foundational diffusion models. Using our framework with a foundational diffusion model allows for comprehensive in-the-wild object-level editing capabilities.

The novelty of our work lies in the way we formulate the image editing tasks that lead to a general approach to **enable comprehensive editing capabilities in various models.** We show the efficacy of our framework on unconditional diffusion models and foundational text-to-image diffusion models. Lastly, we also propose a **multimodal classifier-free guidance** to reap the full benefits of the text-to-image diffusion models. It enables PAIR-Diffusion to control the final output using both reference images and text in a controlled manner hence getting the best of both worlds.

4.5.1.4. Results and outlook

We only present here qualitative results validating that our model can **achieve comprehensive object-level editing capabilities** in practice. Quantitative results and a comprehensive ablation study can be found in our published article (Goel, et al, 2024). We use different baselines according to the editing task. We adapt **Prompt-Free-Diffusion (PFD)** (Xu, et al, 2023) as a baseline for localized appearance editing by introducing masking and using the cropped reference image as input. Moreover, we adopt **Paint-By-Example (PBE)** (Yang, et al, 2023) as a baseline for adding objects and shape editing. For the figures where there is no prompt provided below the image, we assume that the prompt was auto-generated using the template "A picture of *{category of object being edited}*". When editing a local region, we used a masked sampling technique to only affect the selected region (Rombach, et al, 2022).

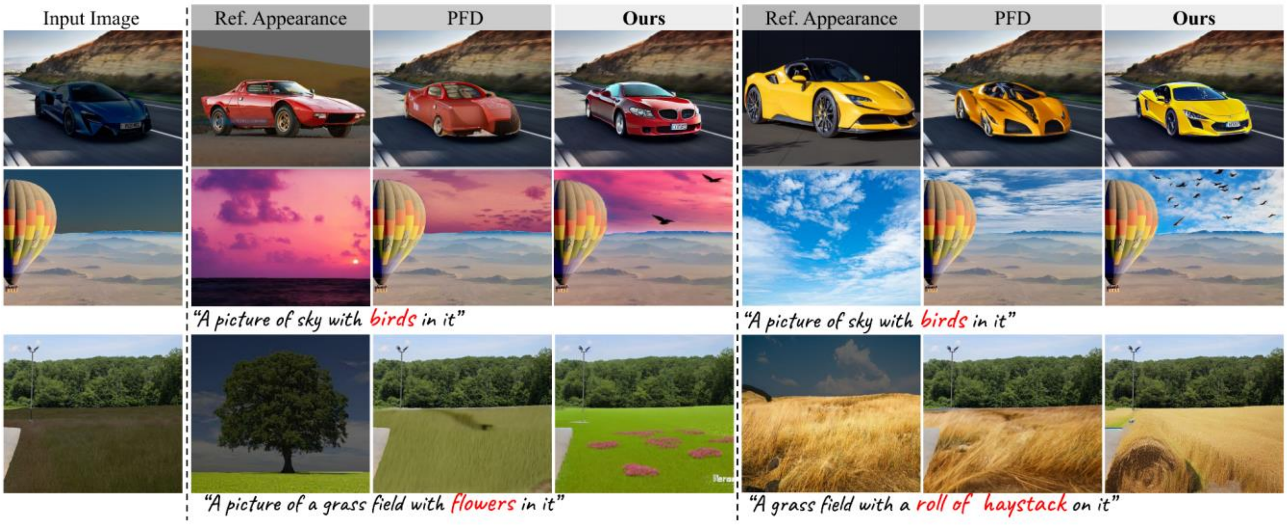

Input Image
Ref. Appearance
PFD
Ours
Ref. Appearance
PFD
Ours
"A picture of sky with birds in it"
"A picture of sky with birds in it"
"A picture of a grass field with flowers in it"
"A grass field with a roll of haystack on it"

Figure 4.5.1.3. Qualitative results for appearance editing

Appearance editing. In Fig. 4.5.1.3, we report qualitative results for appearance editing driven by reference images and text. We can see that our multilevel appearance representation and object-level design help us edit the appearance of both simple objects such as the sky as well as complex objects like cars. On the other hand, PFD (Xu, et al, 2023) gives poor results when editing the appearance of complex objects due to the missing object-level design. Furthermore, using our multimodal classifier free guidance, our model can seamlessly blend the information from the text and the reference images to get the final edited output whereas PFD (Xu, et al, 2023) lacks this ability.

Add objects and shape editing. We show the object addition and shape editing operations result together in Fig. 4.5.1.4. With PAIR Diffusion we can add complex objects with many details like a cake, as well as simpler objects like a lake. When changing the structure of the cake from a circle to a square, the model captures the sprinkles and dripping chocolate on the cake while rendering it in the new shape. In all the examples, we can see that the edges of the newly added object blend smoothly with the underlying image. On the other hand, PBE (Yang, et al, 2023) completely fails to follow the desired shape and faces issues with large objects like lakes.

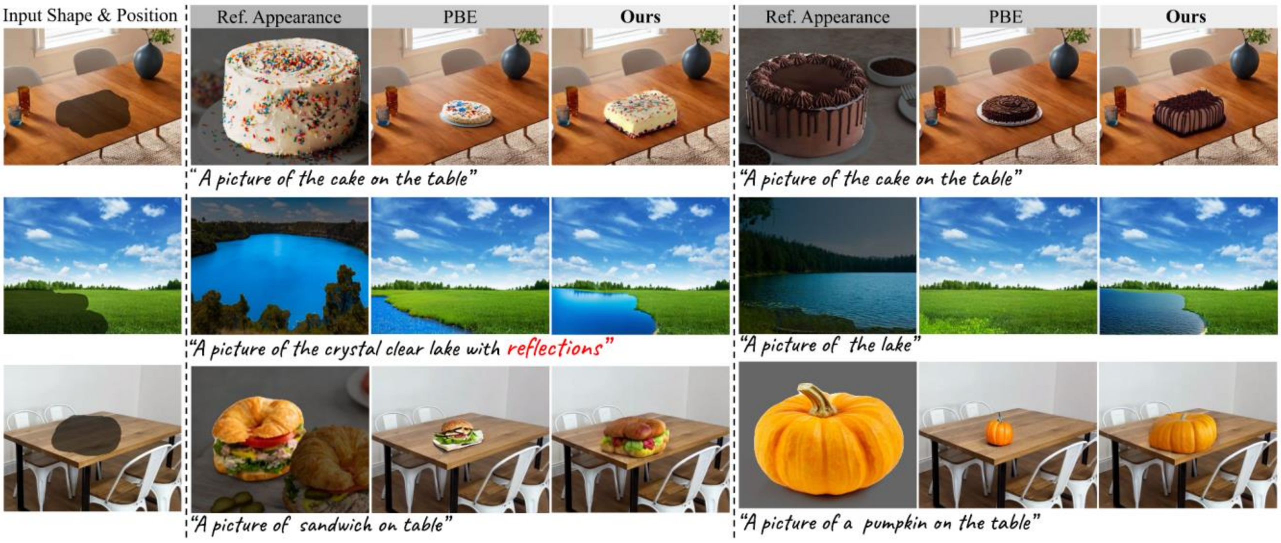

Input Shape & Position
Ref. Appearance
PBE
Ours
Ref. Appearance
PBE
Ours
"A picture of the cake on the table"
"A picture of the cake on the table"
"A picture of the crystal clear lake with reflections"
"A picture of the lake"
"A picture of sandwich on table"
"A picture of a pumpkin on the table"

Figure 4.5.1.4. Qualitative results for adding objects and shape editing.

Limitations and future work. Currently, the architecture modifications present a simple formulation of the appearance vectors and the structure conditioning. While offering advantages by seamlessly integrating into existing Diffusion Models with minimal modification, in the future we plan to explore more sophisticated designs while maintaining the core object-level formulation. We plan to extend the explicit control over other aspects of the objects, such as the illumination, pose, etc., and improve the identity preservation of the edited object. The proposed object-level formulation can also help devise standardizing metrics for image editing tasks in a unified manner, which is lacking in the field.

4.5.1.5. Exposed API for integration

For the moment being, this technology is not exposed through an API. Moreover, we do not foresee integrating this technology in the AI4TRUST platform. Consequently, **exposing this technology through an API will not be necessary**.

4.5.2. Visual Domain Generalization

4.5.2.1. Problem statement

With the development of **deep neural networks** and the introduction of **abundant annotated data**, fully-supervised methods have achieved remarkable success in various visual recognition tasks, including, but not limited to, image classification, object detection, and semantic segmentation. These visual recognition tasks are the fundamental and crucial components of the computer vision world. However, such significant achievements heavily rely on the availability of large-scale annotated data, which are expensive and time-consuming to collect, especially for semantic segmentation and object detection. In addition, even though given abundant labeled training data, the significant performance of the deep learning model is limited to independent and identically distributed (i.i.d.) datasets. Nevertheless, out-of-distribution (OOD) data that are totally unseen during training are inevitably in real-world applications and the models commonly suffer from catastrophic performance degradation when facing unseen situations.

To alleviate the heavy annotation cost and distribution shift**, domain generalization (DG)** (Zhong, et al, 2022) has been introduced in the community. DG only leverages annotated source data to train a robust model that can cope with different unseen conditions. The source domain can be the annotated real-world data but can also be the synthetic data from a pre-designed engine (Richter, et al, 2016), where the latter can greatly reduce the annotation cost. In view of the practicality of DG, previous works have been independently investigating it in **image classification** (Zhou, et al, 2021), **semantic segmentation** (Zhong, et al, 2022), and **object detection** (Wu, et al, 2022). In our research, we aim to propose a unified and versatile framework that is applicable to the above three visual recognition tasks. All these tasks are relevant to AI4TRUST since they allow for **better training of the deep fake classifiers even in the presence of scarce training data** and they make the framework more flexible by being able to generalize to unseen situations.

4.5.2.2. Related work

The **main challenge for DG** is to cope with the significant domain shift between source and unseen target domains, which can be roughly divided into two aspects. First, the diversity in the source data is very limited compared to those of unseen target data. Second, there exists a large distribution gap between the source and target data, e.g., image styles and characteristics of objects. To learn the **domain-invariant model** that can address the domain shift, previous works mainly focus on three aspects: i) designing tailor-made modules (Wu, et al, 2022; Choi, et al, 2021; Pan, et al, 2018) to remove domain-specific information; ii) leveraging extra data to transfer source data (Huang, et al, 2021; Yue, et al, 2019) to possible target styles for narrowing the distribution gap; and iii) diversifying source data within the domain via style augmentation (Zhou, et al, 2021;

Wang, et al, 2019) or adversarial perturbation (Zhong, et al, 2022; Shankar, et al, 2018). However, **the removal of domain-specific information is not complete and explicit** due to the lack of target information; the extra style transfer heavily relies on extra data, which are not always available in practice, and ignores the invariant representation within the source domain. Taking the above into account, we decided to follow the third paradigm to diversify samples in the source domain. In addition, we explicitly introduce **two constraints to help the model effectively learn domain invariant representation and narrow the domain gap**.

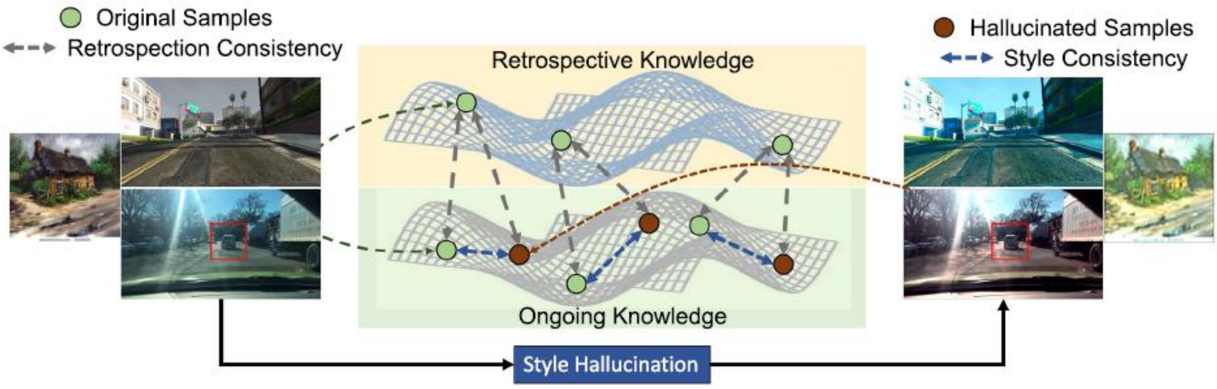


Figure 4.5.2.1. Illustration of the proposed dual consistency constraints for three visual tasks. We generate hallucinated samples (brown circle) from the style hallucination module and then utilize the paired samples and general visual (retrospective) knowledge to learn style consistency (blue dash line) and retrospection consistency (gray dash line).

4.5.2.3. Proposed method

Our proposed approach introduces a **novel dual consistency learning framework (SHADE)** that can jointly address the above two types of domain shift. As shown in Fig. 4.5.2.1, we introduce **two consistency constraints, style consistency (SC) and retrospection consistency (RC)**. SC encourages the model to learn style invariant representation by forcing the consistency between the samples before and after style variation. RC aims to lead the model less overfitting to the source data with the help of general visual knowledge. More details are provided in the original published article (Zhao, et al, 2024).

Specifically, we leverage **the ImageNet (Deng, et al, 2009) pre-trained model** which is available acquiescently in all DG models. The features from the pre-trained model can reflect the representation in the context of the general visual world and thus can serve as the guidance for the ongoing model to retrospect what the visual world looks like and to lead the model less overfitting to the source data. **Style diversifying** is crucial for the success of dual consistency learning, and we adopt the style features, i.e., channel-wise mean and standard deviation, to generate new data. Compared with directly transferring the whole image (e.g., CycleGAN; Zhu, et al, 2017), changing style features can maintain the pixel alignment to the utmost extent, which is better for pixel-level tasks like semantic segmentation.

Previous works (Choi, et al, 2021; Ros, et al, 2016) commonly **mix or swap styles within the source domain**, which will generate more samples of the dominant styles. Nevertheless, it is not the best way since the target styles may be quite different from the dominant styles. To fully take advantage of all the source styles, we propose a **style hallucination module (SHM)**, which leverages C basis styles to represent the style space of C dimension and thus generate new styles. Ideally, the basis styles should be linearly independent so the linear combination of basis styles can represent all the source styles. However, many unrealistic styles that impair the model training are generated when we directly take C orthogonal unit vectors as the basis. To reconcile diversity and realism, we use **Farthest Point Sampling (FPS)** (Qi, et al, 2017) to select C styles from all the source styles as basis styles. Such basis styles contain many rare styles since rare styles are commonly far away from the dominant ones. With these basis styles that represent the style space in a better way, we utilize a **linear combination to generate new styles**.

4.5.2.4. Results and outlook

Datasets. Two synthetic datasets (GTAV (Richter, et al, 2016) and SYNTHIA (Ros, et al, 2016) and three real-world datasets (CityScapes (Cordts, et al, 2016), BDD100K (Yu, et al, 2020) and Mapillary (Neuhold, et al, 2017) are used in our experiments. GTAV (Richter, et al, 2016) contains 24,966 images with the size of 1914×1052, splitting into 12,403 training, 6,382 validation, and 6,181 testing images. SYNTHIA (Ros, et al, 2016) contains 9,400 images of 960×720, where 6,580 images are used for training and 2,820 images for validation. CityScapes (Cordts, et al, 2016) contains 2,975 training images and 500 validation images of 2048×1024. BDD100K (Yu, et al, 2020), and Mapillary (Neuhold, et al, 2017) contains 7,000 and 18,000 images for training, and 1,000 and 2,000 images for validation, respectively.

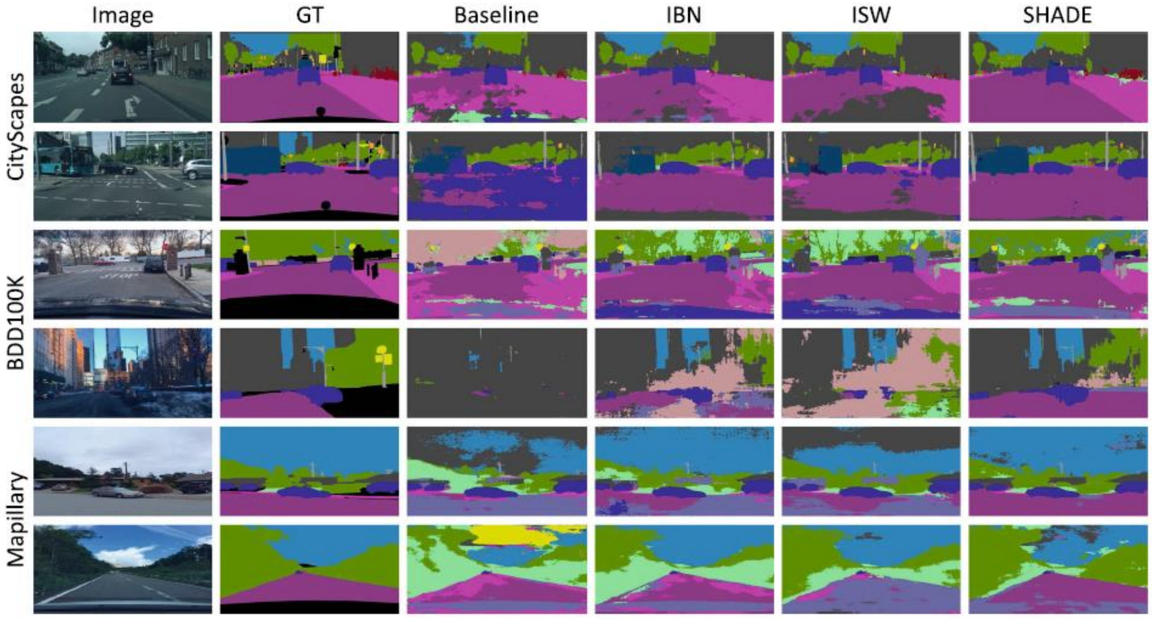


Figure 4.5.2.2. Qualitative comparison of segmentation results.

Qualitative results. To demonstrate the effectiveness of SHADE, we compare the qualitative results of semantic segmentation and object detection. We compare the segmentation results among baseline, IBN-Net (Pan, et al, 2018), ISW (Wang, et al, 2019) and SHADE on CityScapes, BDD100K and Mapillary in Fig. 4.5.2.2. We obtain two observations from Fig. 4.5.2.2. First, SHADE consistently outperforms other methods under different target conditions (e.g., sunny, cloudy, and overcast). Second, SHADE can well deal with both background classes (e.g., road) and foreground classes (e.g., bus and bicycle).

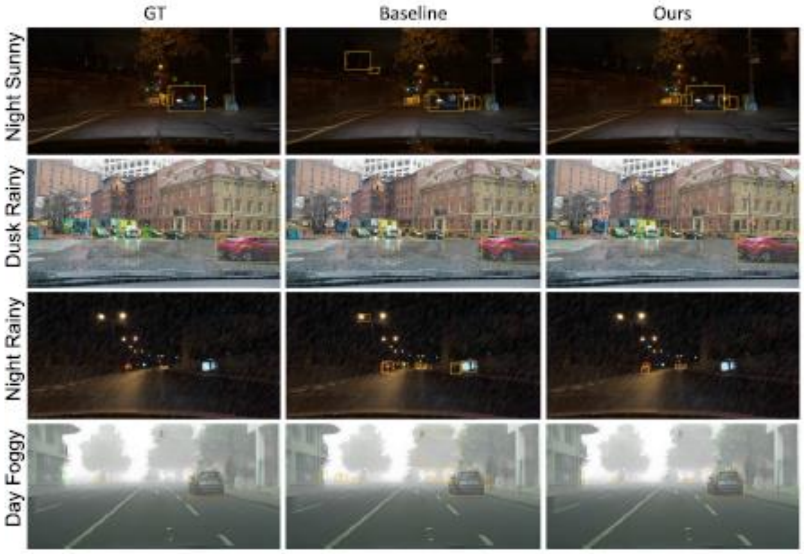


Figure 4.5.2.3. Qualitative comparison of object detection results.

We also compare SHADE with the baseline model on object detection benchmark in Fig. 4.5.2.3 and SHADE consistently outperforms the baseline under different environmental conditions. The above observations demonstrate that SHADE is robust to style variation and has strong ability in addressing unseen images.

Final discussion. SHADE is an effective and versatile framework which can be applied to image classification, semantic segmentation and object detection tasks with both ConvNets and Transformer backbone and can achieve state-of-the-art performance on different benchmarks and under different settings. To address the distribution shift between the source and unseen target domains, SHADE leverages two consistency constraints to learn the domain invariant representation by seeking consistent representation across styles and the guidance of retrospective knowledge. In addition, the style hallucination module (SHM) is equipped into our framework, which can effectively catalyze dual-consistency learning by generating diverse and realistic source samples.

4.5.2.5. Exposed API for integration

For the moment this technology is not exposed through an API. Moreover, we do not foresee integrating this technology in the AI4TRUST platform. Consequently, **exposing this technology through an API will not be necessary.**

5. Multimodal data analysis methods

5.1. Video anomaly detection

Delving into CLIP latent space for Video Anomaly Recognition

5.1.1. Problem statement

We tackle the complex problem of **detecting and recognizing anomalies in videos** at the frame level, utilizing only **video-level supervision**. We introduce the novel method **AnomalyCLIP**, the first to combine Large Language and Vision (LLV) models, such as CLIP, with multiple instances learning for joint video anomaly detection and classification. Our approach specifically involves manipulating the latent CLIP feature space to identify the normal event subspace, which in turn allows us to effectively learn text-driven directions for abnormal events. When anomalous frames are projected onto these directions, they exhibit a large feature magnitude if they belong to a particular class. We also introduce a **computationally efficient Transformer architecture** to model short- and long-term temporal dependencies between frames, ultimately producing the final anomaly score and class prediction probabilities. We compare AnomalyCLIP against state-of-the-art methods considering three major anomaly detection benchmarks, i.e. ShanghaiTech, UCF-Crime, and XD-Violence, and empirically show that it **outperforms baselines in recognizing video anomalies.**

5.1.2. Related work

Video anomaly detection (VAD) methods can be categorized into fully-supervised (Bai et al., 2019; Wang et al., 2019), weakly-supervised (Li et al., 2022a,b; Sultani et al., 2018; Tian et al., 2021; Wu and Liu, 2021), one-class classification (Liu et al., 2021; Lv et al., 2021; Park et al., 2020), and unsupervised approaches (Narasimhan, 2018; Zaheer et al., 2022). Weakly-supervised methods, requiring only video-level annotations, have gained popularity, as they typically yield good results while limiting the annotation effort. Sultani et al. (2018) were the first to formulate weakly-supervised VAD as a multiple-instance learning (MIL) task, dividing each video into short segments that form a set, known as bag. Bags generated from abnormal videos are called positive bags, and those generated from normal videos negative bags. Since this pioneering work, MIL has become a paradigm for VAD and several subsequent works have proposed to refine the associated ranking model to predict anomaly scores more robustly. Whilst existing weakly-supervised VAD methods have shown to be effective in anomaly detection (Li et al., 2022a), they are not designed for recognizing anomaly types (e.g., shooting vs. explosion).

The emergence of **novel Large Language and Vision (LLV)** (Radford et al., 2021; Schuhmann et al., 2021, 2022; Singh et al., 2022), which can learn joint visual-text embedding spaces, has enabled unprecedented results in several image and video understanding tasks (Xu et al., 2021; Wang et al., 2021). In AI4TRUST, we introduce **the first method that jointly addresses VAD and VAR with LLV models.**

5.1.3. Proposed method

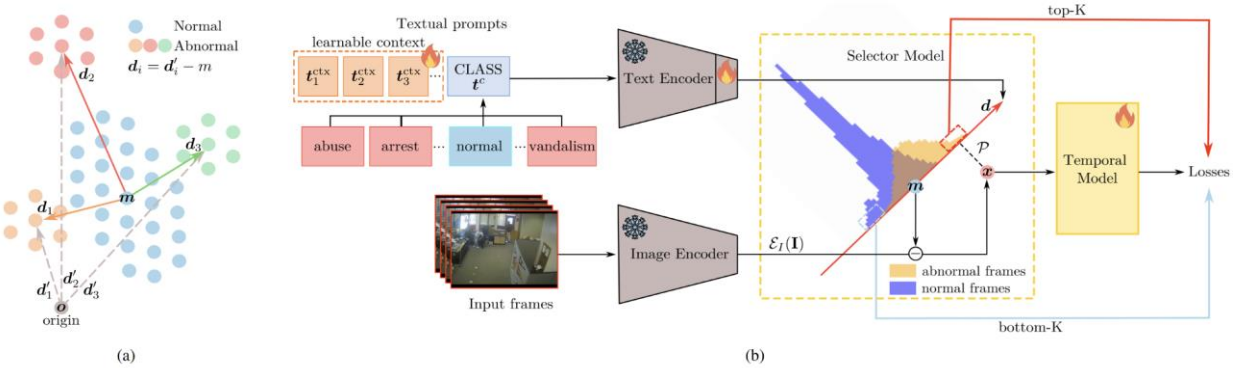


Fig. 5.1.1. (a) Illustration of the CLIP space and the effects of the re-centring transformation with features of normal. (b) Illustration of our proposed framework.

We propose to leverage the **CLIP model** (Radford et al., 2021) to address VAR and show that: i) the alignment between the visual and textual modalities in the CLIP feature space can be used as an effective likelihood estimator for anomalies; ii) such estimator, not only can detect anomalous occurrences, but also their types; iii) such estimator is effective only when adopting our proposed CLIP space re-centering transformation (see Fig. 5.1.1. (a)). Our method is composed of two models as shown in Fig. 5.1.1. (b): a **Selector model** and a Temporal model. The Selector model *S* produces the likelihood that each frame belongs to an anomalous class *S(x)* ∈ R^C, where C is the number of anomalous classes. We exploit the vision-text alignment in the CLIP feature space and the CoOp prompt learning approach (Zhou et al., 2022) to estimate this likelihood. The Temporal model *T* assigns a binary likelihood to each frame of a video indicating whether the frame is anomalous or normal. Unlike *S, T* exploits temporal information to improve predictions and we implement it with a Transformer network (Ho et al., 2019). The predictions from *S* and *T* are then aggregated to produce a distribution indicating the **probability of a frame being normal or abnormal**, and which abnormal class it belongs to. We train our model using a combination of MIL and regularization losses. Importantly, as *T* is randomly initialized, the likelihood scores are less reliable, thus we always use the likelihoods produced by *S* to perform segment selection in MIL.

5.1.4. Results and outlook

Datasets. We perform our study using three widely-used VAD datasets, i.e., ShanghaiTech (Liu et al., 2018), UCF-Crime (Sultani et al., 2018), and XD-Violence (Wu et al., 2020). ShanghaiTech consists of 437 videos, recorded from multiple surveillance cameras on a university campus. A total of 130 abnormal events of 17 anomaly classes are captured in 13 different scenes. UCF-Crime is a large-scale dataset of real-world surveillance videos, containing 1900 long untrimmed videos that cover 13 real-world anomalies with significant impacts on public safety. XD-Violence is a large-scale violence detection dataset comprising 4754 untrimmed videos with audio signals and weak labels, divided into a training set of 3954 videos and a test set of 800 videos.

We test our model on surveillance data due to their availability and fair comparison with the state-of-the-art. The final version of the model will be tailored to anomalies relevant for the project objectives (i.e. events that are completely unexpected to appear in video) regardless of whether they are recorded by surveillance cameras or other sources.

Quantitative results. We compare AnomalyCLIP against state-of-the-art methods. As no previous method address the VAR task, we produce baselines by repurposing some best-performing VAD methods including RTFM (Tian et al., 2021), S3R (Wu et al., 2022) and SSRL (Li et al., 2022a), and CLIP-based baselines (Radford et al., 2021; Wang et al., 2021).

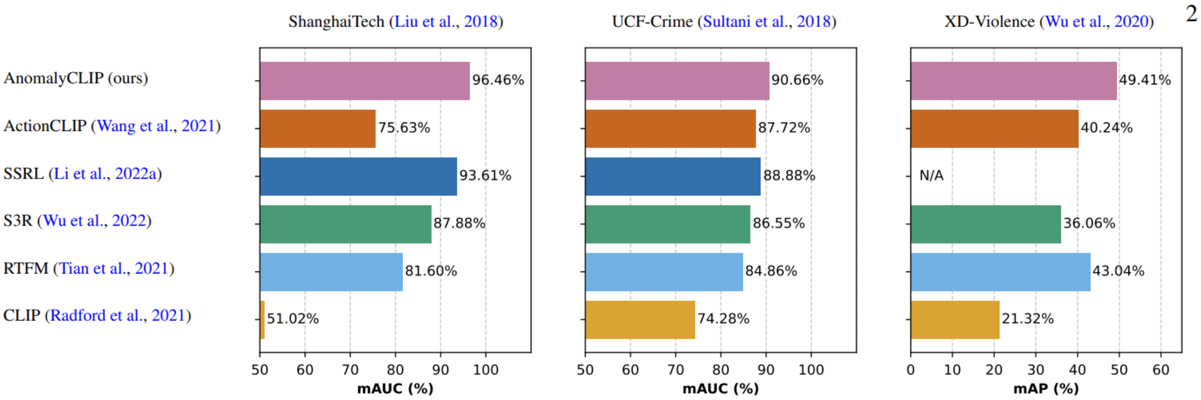


Fig. 5.1.2. Comparison of various anomaly recognition methods on the ShanghaiTech, UCF-Crime, and XD-Violence datasets in terms of the mean area under the curve (mAUC) of the receiver operating characteristic and the mean average precision (mAP) of the precision-recall curve.

As shown in Fig. 5.1.2, AnomalyCLIP obtains state-of-the-art VAR results in mAUC and mAP. For additional quantitative results and a comprehensive ablation study please refer to our article (Zanella et al., 2023).

Qualitative results. Fig. 5.1.3 presents the qualitative results of our proposed AnomalyCLIP in detecting and recognizing anomalies within a set of UCF-Crime, ShanghaiTech, XD-Violence test videos. For each video, we show at the bottom the predicted probability of each frame being anomalous by our model over the number of frames. We showcase some key frames to reflect the relevance between the predicted anomaly probability and the visual content. The red-shaded areas denote the temporal ground-truth of anomalies. The model can predict both the presence of anomalies in test videos and the category of the anomalous event. We also indicate the predicted anomalous class for detected abnormal frames in the red boxes, while videos without detected anomalies are indicated with blue boxes as Normal.

Future work. As future work, we foresee extending our approach to address the challenge of open-ended video anomaly detection. This involves developing methods to automatically identify and classify anomalous events or behaviors in videos without relying on predefined categories or labels.

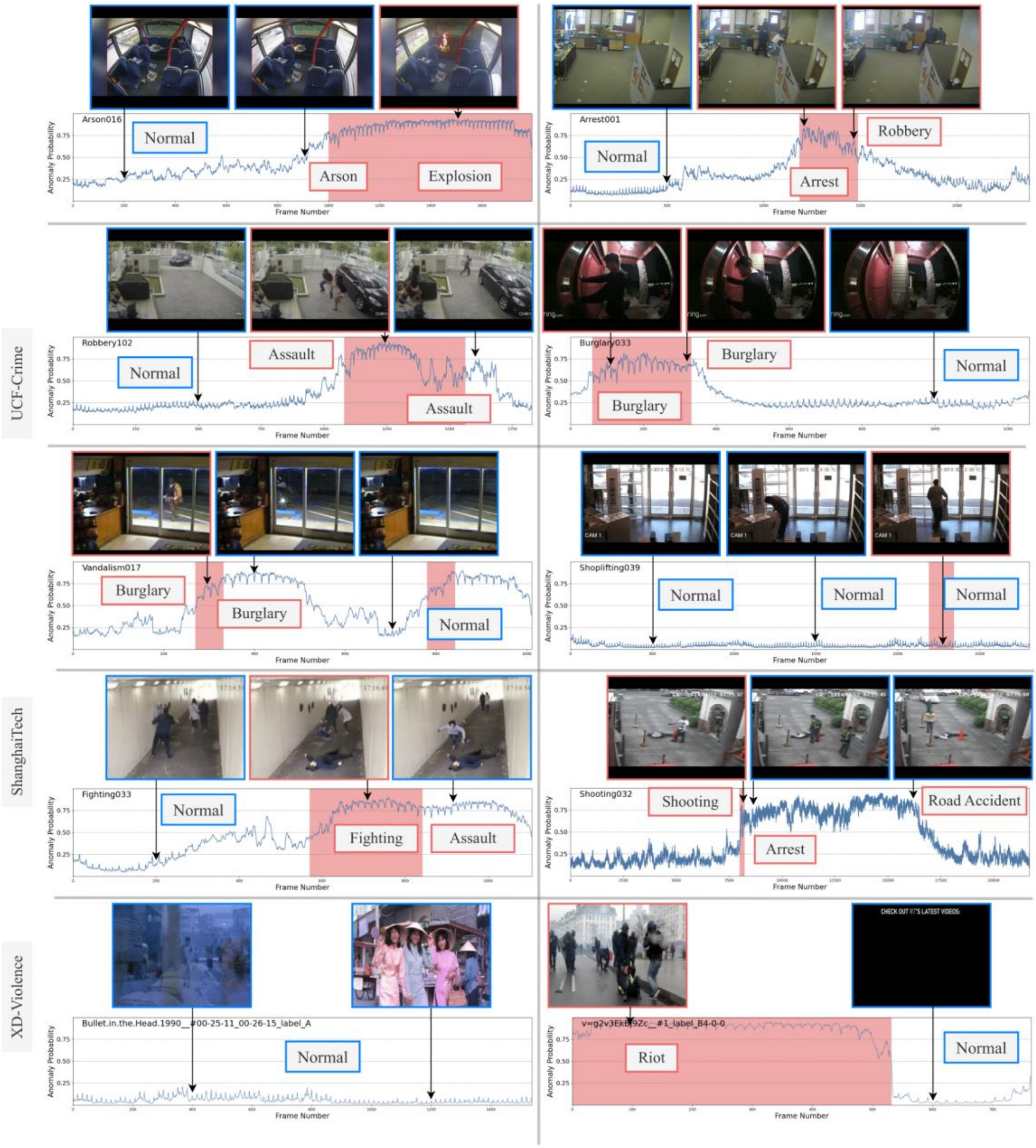


Fig. 5.1.3. Qualitative results for VAR.

5.1.5. Exposed API for integration

For the moment, **this technology is not exposed through an API**. This will be done in the future to allow its integration with the AI4TRUST platform and the Disinformation Warning System.

5.2. Audio anomaly detection

5.2.1. Problem statement

Anomaly detection aims to identify out-of-distribution samples. For deepfake detection, this means modeling the distribution of *all* real samples, and then anomalies would ideally correspond to deepfakes. This task is **challenging** because the distribution of real samples is incredibly broad: music, ambient sounds, speech in Yorùbá, all can be real, but they might look like anomalies if our source distribution is English speech data.

To make progress on this task, we focus on ***local* anomalies**; we assume that most of the input signal is real, but there are local alterations. For example, these alterations could be short fake segments or splice points (i.e., locations where two real files have been concatenated). While small, these manipulations can completely change the meaning of a sentence; consider inserting the word “not” in a sentence or, conversely, deleting it.

5.2.2. Related work

We consider two main directions for audio anomaly detection with respect to the audio deepfake field: (i) ***audio splicing detection***, i.e., identifying regions in the audio signal which have been replaced/deleted/inserted using a different real audio source; and (ii) ***partial spoofing detection***, *i.e.*, identifying regions in the audio signal which have been replaced/deleted/inserted from a fake audio source.

Within the field of **audio splicing**, the traditional approaches relied mainly on the background signal information, such as noise (Pan et al, 2012) or reverberation (Zhao et al., 2017, Capoferri et al., 2020). Handcrafted representations (such as MFCCs) of the signal are fed to the splicing detection algorithms, and changes within this background content across the input are used to indicate a potential tampering of the signal. The deep-learning based methods remove the handcrafted representations, and use the raw signal or learnable representations as input. For example, Jadhav et al. (2019) and Zhang et al. (2022a) use convolutional neural networks to identify tampered samples. Zhang et al. (2022a) and Zhang et al. (2022b) add the separate task of localizing the splicing point rather than just performing a detection over the entire signal. The idea of exactly localizing splicing points is also adopted by Moussa et al. (2023) where the network is also tailored towards unconstrained splicing detection, meaning that splices can occur at any point and under various conditions (e.g., background noise, compression artifacts).

While most of the research on audio deepfakes focuses on identifying utterances which were entirely generated by a TTS or VC method, a new topic has recently emerged. It is called **partial spoofing** and refers to the manipulation of only a subsegment of the original carrier. As a result, the task of the solving algorithms is not only to identify if a sample has been manipulated, but also to mark the exact location where the manipulation took place - similar to the audio splicing localisation method. One of the first studies and datasets of this field was that of Zhang et al. (2022c) which introduces the PartialSpoof database, but also shows a series of results using light fully connected, recurrent or gMLP architecture over Self Supervised Learning (SSL)-derived representations, aimed at generating frame-level spoofing scores. Cai et al. (2022) describe a series of results for audio and video deepfake localisation using separate encoders for each modality. The audio part is converted into spectrograms or Mel frequency cepstral coefficients and ran through a multi-layered 2D CNN network.

The network then uses **three loss functions** to combine the input representations and output the frame-level predictions. Xie et al. (2023) add a temporal deepfake location module aimed at providing timestamps of the spoofed segments. Additionally, SSL-derived embeddings are used to measure the similarity between the fake and real segments. A combination of frame- and utterance-level representations is used by Khan et al. (2023) to identify tampered frames from a learnable spectro-temporal representation. LSTM and biLSTM modules are employed as the learning modules.

5.2.3. Proposed method

Given an audio file, we want to **automatically localize the places where it has been manipulated**. As such, the output will be a list of labels (“fake” or “real”) for all audio segments, which cover the entire input audio. These segments correspond to short non-overlapping windows of size 20 ms or 160 ms. A segment should be labeled as "fake" if *any* part of the audio in the window has been manipulated. This formulation follows the setup proposed in the PartialSpoof dataset (Zhang et al., 2022c) on which we base our experiments.

We assume a fully supervised learning scenario, where we have access to samples with local manipulations, as well as the corresponding local labels. We formulate this task as a sequence-to-sequence problem. In particular, we **consider three types of architectures: convolutional networks, gated multi-layer perceptrons** (gMLP; Liu et al., 2021), and **Transformer** (Vaswani et al., 2017). We train these methods using the frame-wise cross-entropy loss. All models rely on the same backbone features: the wav2vec 2.0 XLS-R 2B representation, which we have shown to yield best results (see Section 3.2.4).

5.2.4. Results and outlook

We carried our experiments on the **PartialSpoof database** (Zhang et al., 2022c). This dataset contains two types of local manipulations: insertion of fake segments and splicing - concatenation of real segments. These two types of manipulations are not differentiated in the dataset. The dataset is derived from ASVspoof'19, so the real samples are from the VCTK database[44] and fake segments are synthetized with methods from the ASVspoof’19 database (19 systems including both text-to-speech and voice conversion methods). The dataset amounts to around 120 hours, with an average duration of the audio files of around three seconds.

	method	frontend	backend	train window (ms)	window test (ms)	
					20	160
1	Zhang et al. (2023)	wav2vec2-large	gmlp	{20, 40, 80 ... 640}	12.9	9.2
2	Xie et al. (2024)	wav2vec2-xls-r	TDL	160		7.0
3	ours	wav2vec2-xls-r-2b	linear	20	14.8	28.3
4	ours	wav2vec2-xls-r-2b	linear	160	22.3	12.8
5	ours	wav2vec2-xls-r-2b	conv	20	**8.1**	**6.0**
6	ours	wav2vec2-xls-r-2b	gmlp	160		12.8
7	ours	wav2vec2-xls-r-2b	transformer	160		10.0

Table 5.2.1. Equal error rate (EER) on the PartialSpoof test dataset. All models are trained on the PartialSpoof train split.

Table 5.2.1 shows the results for the proposed methods (rows 3–7) and in comparison with state-of-the-art approaches (rows 1–2). We considered **two types of windows for this evaluation: a fine-grained variant, of 20ms, and a coarser one of 160ms, which is more typical in the literature.** Among the methods tested, the convolutional network achieves the best results for both test window sizes: 8.1% EER at window size of 20ms and 6.0% EER at window size of 160ms. **This performance is better than state-of-the-art**, which can be attributed to the improved front-end features (wav2vec2-xls-r-2b). When the front-end features are kept fixed, we observe that more flexible back-end methods, such as gMLP or Transformer, do not help; in fact, these two are not much better even than the simple linear layer. An explanation could be related to overfitting, but we still need to confirm this hypothesis. By training on various window sizes (rows 3 and 4), we estimate their impact on performance. We observe that **the train-test window mismatch increases the error**: from 14.8% to 22.3% EER for the 20ms test window size; from 12.8% to 28.3% EER for the 160ms test window size.

[44] https://datashare.ed.ac.uk/handle/10283/2950 (accessed: 2023-03-22)

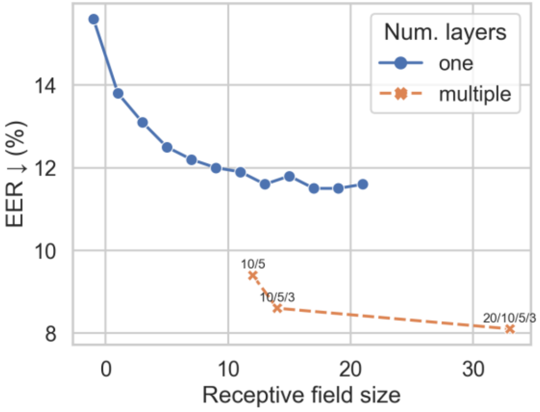


Figure. 5.2.1. Equal error rate (%) of the convolutional neural network model as a function of the receptive field and number of layers. For the multi-layer networks (orange curve), a notation of the form m/n/.../p indicates the kernel size for each layer. The muli-layer networks use ReLU activations between the layers.

To better understand our best-performing model, the **convolutional network**, we performed a **sensitivity analysis**. We monitored performance in terms of two factors: the network's receptive field and its expressivity. The receptive field, i.e., how much of the input is used to make a prediction for a single frame, is controlled by the kernel size and the number of layers. In terms of expressivity, we distinguished between single-layer networks (which are linear models) and multi-layer networks (which are non-linear models because we intersperse ReLU non-linearities between the layers). The results shown in Figure 5.2.1 highlight the importance of both components: the single-layer network (blue curve) improves significantly (from 15.6% to 11.6% EER) with the kernel size, which equates the receptive field; the non-linear multi-layer variant (orange curve) improves significantly over the linear variant at the same receptive field (from 11.6% to 8.6% EER). After a certain point, however, the gains diminish; more than doubling the receptive field and adding a layer brings only 0.5% EER improvement (the difference between the rightmost two orange points).

Our next step is to **validate our approach on challenging out-of-domain datasets**, such as LAV-DF (Cai et al., 2022), which consists of partially manipulated videos, or HalfTruth (Yi et al., 2021), which consists of partially manipulated audio files, but in a different language, Chinese.

5.2.5. Exposed API for integration

This technology will be **exposed in a future version of the AI4TRUST platform**.

5.3. Visual-text misalignment detection

5.3.1. Problem statement

One type of disinformation is based on the **use of an image/video under a different context** (than the original one) to mislead the viewers and cause false impressions. An example of such a fake was used to support a conspiracy theory about the health of Mrs. Clinton during her campaign for the presidential elections of 2016 in the USA[45]. An image/video showing Mrs. Clinton slipping as she walks up the stairs into a residential facility was used as evidence that she is suffering from seizures. This is just an example, since misalignment between the visual content and its textual description could be found in various forms, including inaccuracies, inconsistencies, or mismatches between the depicted objects/scenes in the image/video and the textual description. To spot cases that are worthy of verification, we need **technologies for visual-text misalignment detection**. Such technologies would allow to identify cases where the textual content accompanying an image/video is not relevant to the depicted events or subjects, since such cases could potentially be used to mislead the viewers about an event.

5.3.2. Related work

The widespread use of social media and the Internet made it possible to spread news, or information in general, fast and to a wide audience. This advancement, however, also gave the opportunity to **spread disinformation to affect the public opinion about various topics**. This emerging problem is gathering increasing attention from researchers who explore ways to tackle this situation. Early studies were exploring methods to identify false claims either through text analysis (Mridha et al., 2021) or based on the detection of manipulated images (Rana et al., 2022). However, news websites nowadays include images and/or videos in their articles since readers are more attracted to multimedia content (Li & Xie, 2020). The presence of such content can make the story more convincing, whether it is true or not (Newman et al., 2012). This advancement pointed out **the importance of multimodal misinformation detection (MMD)**, which relates to the process of identifying false or misleading information using multiple forms of data. The recent proliferation of multi-modal large language models (LLMs) gave the opportunity to address this task effectively. Consequently, several datasets have been created for training network architectures to distinguish real from deceiving claims. Most of these datasets are extensions of the VisualNews dataset (Liu et al. 2021), which contains real pairs of images and associated captions from the news domain. For example, RSt and CSt (Papadopoulos et al., 2023) were created after performing random sampling by topic and CLIP-based sampling by caption-to-caption similarity, respectively.

[45] Makela, M. (2016, February 24). *Hillary Clinton Campaigns In South Carolina Ahead Of Primary*. Getty Images. https://www.gettyimages.com/detail/news-photo/democratic-presidential-candidate-former-secretary -of-state-news-photo/512026552

CLIP-NESt and R-NESt were formed after applying in-topic CLIP-based and random-based entity swapping, respectively (Papadopoulos et al., 2023). **NewsClippings++** was formulated by decontextualizing image-text pairs using the multimodal encoder CLIP, as well as scene and person matching computer vision models, (Luo et al. 2021) while a subset of this dataset (denoted as NC-t2t in Papadopoulos et al., 2024) was formulated by defining misaligned pairs through text-to-text similarity. On a slightly different basis, the **CHASMA dataset** (Papadopoulos et al., 2024) and its **CHASMA-D variant** with more balanced classes after applying random down-sampling, were developed after employing a large pre-trained cross-modal alignment model (CLIP) to pair legitimate images from VisualNews, with contextually relevant but misleading texts from the Fakeddit dataset (Nakamura et al., 2020). **Fakeddit** is a large weakly labeled dataset consisting of several instances collected from various subreddits[46] and grouped into a number of classes based on their content. The **MEIR dataset** (Sabir et al., 2018) has been proposed mainly to support image repurposing detection, and thus contains pairs of image-text where the text was changed after location, person, and organization manipulations on real-world data sourced from Flickr. The **VERITE dataset** (Papadopoulos et al., 2024) was formed by collecting image and text data from fact-checked articles from Snopes and Reuters that were classified as “MisCaptioned”. Finally, the **COSMOS dataset** (Aneja et al., 2021 contains real-world multimodal misinformation; namely it consists of 1,700 image-text pairs and is balanced between truthful and misleading ones (collected from credible news sources and Snopes.com respectively).

5.3.3. Proposed method

Based on the literature review, we initially aimed **to create a training dataset** that is well-tailored to the needs of our task. In particular, instead of having examples of mis-captioned images, where the caption can be highly irrelevant to the visual content (as in many of the existing dataset), we worked on building a dataset with pairs of image-text that fit more to our needs for detecting cases where the visual content was used out of its original context. For this, we utilized the **VisualNews dataset and the Phi-2[1] Large Language Model** to generate misaligned (and thus misleading) versions of real captions accompanying images. For example, our methodology takes the caption “*Votes against environmental laws by the Republican Dominated Congress*” and produces the following misleading one: “*Votes in favor of environmental laws by the Republican Dominated Congress.*”. Through this methodology, we created the VisualNewsDC (de-contextualized) dataset that is double in size (approx. 2M image-text pairs) compared to the VisualNews dataset and contains a balanced amount of genuine and misaligned image-text pairs. To assess the usefulness of this dataset, we used it to train an **existing method from the literature** (Papadopoulos et al., 2024) and **evaluate its performance on the COSMOS dataset**.

In addition, we developed a new network architecture for visual-text misalignment detection. This architecture consists of a visual and a text encoder and is trained using a triplet loss function. The

[46] https://www.reddit.com

visual encoder utilizes a **vision transformer (ViT) architecture** to extract high-level visual features from input images. The ViT model is fine-tuned on the CLIP task to align images with their corresponding captions. The text encoder employs a transformer-based architecture to encode textual descriptions into embedding vectors. This encoder is also pre-trained on CLIP's objective, thus allowing the creation of a joint multimodal embedding space. The triplet loss function is designed to train the network architecture by minimizing the distance between the images (anchor) and the corresponding real captions while maximizing the distance between real and misleading captions. For training the developed network architecture, the data samples of the created VisualNewsDC dataset are fed into a vision-language pre-trained model, specifically **an openCLIP variation of the Contrastive Language-Image Pretraining (CLIP) model** (Radford et al., 2021), and the extracted features are given as input in the encoders of the architecture. Each training sample consists of an anchor (the image), a positive example (the genuine caption) and a negative example (the misleading caption). Based on the employed triplet loss, the model learns to distinguish between genuine and misleading texts associated with the same visual content, thereby enhancing its ability to detect misalignments between an image and the accompanying text.

5.3.4. Results and outlook

For performance evaluation, we used the **COSMOS dataset.** This dataset consists of images and captions scraped from news articles and other websites designed for training and evaluation of out-of-context use of images. It is divided **into three splits: Training (160 K images), Validation (40 K images) and Test (1700 images).** For training, out-of-context annotations do not exist. The test set was manually annotated via an in-house annotation tool from the creators of the benchmark. The statistics of the dataset as well as some indicative examples of image-text misalignment are given in Table 5.3.1 and Fig. 5.3.1.

Split	# Images	# Captions	Context Annotation
Train	161,752	360,749	No
Valid	41,006	90,036	No
Test	1,700	3,400	Yes

Table 5.3.1: Statistics of the COSMOS dataset

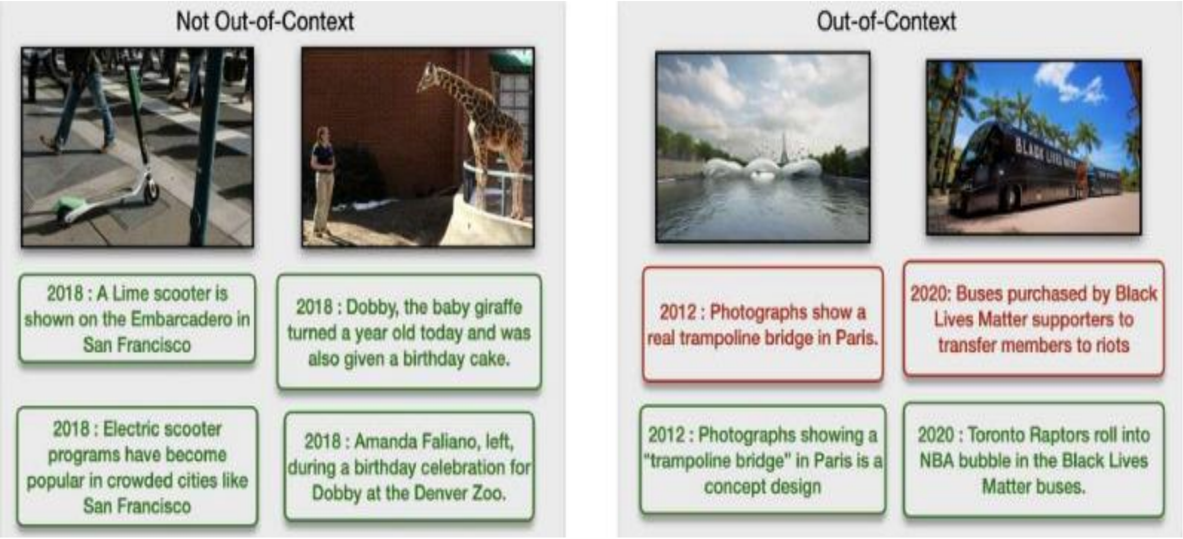

Not Out-of-Context
2018 : A Lime scooter is shown on the Embarcadero in San Francisco
2018 : Dobby, the baby giraffe turned a year old today and was also given a birthday cake.
2018 : Electric scooter programs have become popular in crowded cities like San Francisco
2018 : Amanda Faliano, left, during a birthday celebration for Dobby at the Denver Zoo.
Out-of-Context
2012 : Photographs show a real trampoline bridge in Paris.
2020: Buses purchased by Black Lives Matter supporters to transfer members to riots
2012 : Photographs showing a "trampoline bridge" in Paris is a concept design
2020 : Toronto Raptors roll into NBA bubble in the Black Lives Matter buses.

Fig. 5.3.1: Examples from the COSMOS dataset where images from social media and online news were used out of context (right) and those which were not (left). (Red) denotes false captions and (green) shows the true captions along with year published.

The results of our evaluations about the usefulness and suitability of the created VisualNewsDC dataset for training a visual-text misalignment detection method are present in Table 5.3.2. These results show that our dataset allows the D (I, C) method of (Papadopoulos et al. 2024) to learn a better modeling for distinguishing aligned from misaligned pairs of image-text, and **achieve higher performance on the COSMOS dataset**.

Model	Training Dataset	Accuracy
D (I, C)	RSt	51.5
D (I, C)	NC-t2t	52.6
D (I, C)	CSt	52.2
D (I, C)	MEIR	53.2

Model	Training Dataset	Accuracy
D (I, C)	CLIP-NESt	53.4
D (I, C)	R-NESt	55.2
D (I, C)	Fakeddit	52.5
D (I, C)	CHASMA	58.9
D (I, C)	CHASMA-D	61.8
D (I, C)	VisualNews-DC (ours)	**63.2**

Table 5.3.2. Results of the D(I, C) model on the COSMOS benchmark.

Based on this finding, we utilized the created **VisualNewsDC dataset** to train our network architecture and compare its performance with the one from (Papadopoulos et al., 2024). The performance comparison reported in Table 5.3.3 indicates that our method is able to obtain a better understanding of the task, as it exhibits noticeably higher performance on the COSMOS dataset.

Model (Inputs)	Training Data	Accuracy
D (I, C)	VisualNews-DC (ours)	63.2
openCLIP - ViT-H-14 - FT (I, C) (Ours)	VisualNews-DC (ours)	**68.9**

Table 5.3.3. Results on the COSMOS benchmark.

In the next months, we will experiment with additional datasets and approaches for generating training data and explore the use of different backbone models and training approaches for **further improving the performance of the current visual-text misalignment detection method**.

5.3.5. Exposed API for integration

For the moment this technology is not exposed through an API. However, we plan to deploy the best-performing model in REST-based API that will be exposed to assist the integration of this technology in the AI4TRUST platform and the Disinformation Warning System.

5.4. Multimodal deepfake video detection

5.4.1. Problem statement

Deepfakes are a popular multimedia type that emerged recently and has flooded the digital landscape (see Section 1.3.2 of deliverable D2.1 of WP2 for a thorough description of this technology and its societal implications). In addition to being used for entertainment, **deepfakes have been widely used in malicious ways**, such as impersonation, and pose new challenges to cyber security and privacy. Several pertinent datasets and detection methodologies have been proposed; however, most of them tackle the unimodal case in which either a video, an audio, or an image is manipulated. The **combination of manipulated video and audio** is much more challenging to be detected and needs specialized tools to be addressed.[47]

5.4.2. Related work

We experiment with improvements on the work of Chugh et al. (2020), which investigates **bimodal deepfake detection modeling pipelines** based on the idea of the dissonance of audio and visual parts of manipulated videos, i.e., the notion that since deepfake manipulations have either the visual or audio components tampered, but not both, the audio and visual channels are at each video time-step out-of-sync. The modeling based on the aforementioned hypothesis is implemented by applying the contrastive loss on the difference of the high-level semantic facial feature representations (extracted from second-to-last fully connected layer) of the input video and audio streams to capture the modality dissonance score.

5.4.3. Proposed method

Network architecture

The model architecture consists of **two branches**, one for the processing of the **video** and one for the processing of the **audio** modality, as illustrated in Figure 5.4.1. The visual branch extracts video segments of 1 second duration with 30 frames per second, if faces are detected using the S3FD face detector (Zhang et al. 2017). Segments in which S3FD does not detect faces are discarded from the training set. The resulting video segment's collated tensor of size (3, 30, 224, 224) is fed to a Resnet CNN classifier. The specific architecture used is the 3D Resnet-18 inspired by (Hara et. al. 2018), consisting of 3D Convolutional layers and Residual blocks. The audio branch extracts audio segments of 1 second duration from each video, using the ffmpeg library[48], for a maximum of 10 segments per input video due to storage restrictions. The resulting .wav files (segments) are

[47] https://link.springer.com/article/10.1007/s10489-022-03766-z#Sec5
[48] https://ffmpeg.org/

then read and converted to mono channel using librosa[49]. For audio feature representation the following two methods are considered:

- The Mel Frequency Cepstral Coefficients (MFCCs) of each audio segment's waveforms are computed for 13 distinct Mel frequency bands and superimposed to determine the segment-based video spectrogram of dimensionality 10 * 18 * 1 / f-audio, where f-audio ~ 22050 Hz corresponds to the audio sampling rate and varies in relation to input audio segment. The temporally-averaged segment MFCCs are then superimposed for each segment, yielding the 10s spectrogram with shape 1x18x10 that is used as input to a CNN encoder architecture for feature extraction.
- Utilizing pretrained end-to-end speech models, deep features are extracted after fine tuning on audio signals from extracted audio streams from the deepfake datasets.

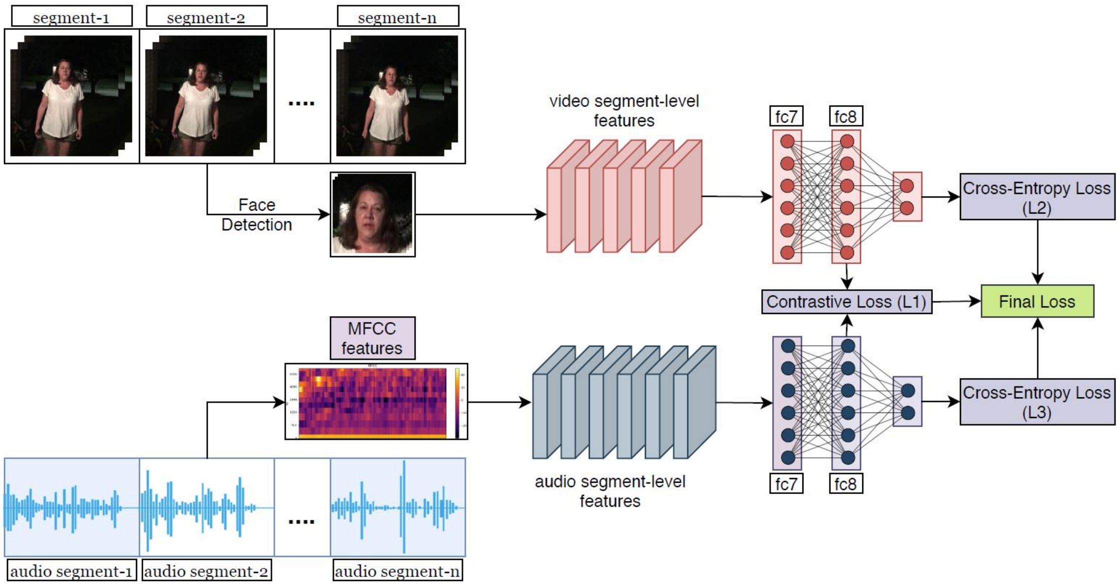


Figure 5.4.1. The main components of the audio-visual Deepfake detection methodology that we adopted from (Chugh et al. 2020).

Used data

We considered videos from **DeepFake Detection Challenge (DFDC)**, using a 85:15 train:test split. The DFDC audio component is in most videos not manipulated, therefore the Dissonance assumption can be applied effectively. The training samples were preprocessed by splitting them in 10 distinct 1-second intervals. Also, we considered the **FakeAVCeleb** (Khalid et al. 2021) datasets for experimentation wrt audio feature extraction.

[49] https://github.com/librosa/librosa

Implementation details

Image segments were normalized by ImageNet 1k statistics to improve the model robustness. Similarly, **MFCC features were preprocessed with the Cepstral Mean and Variance Normalization (CMVN) method** to reduce distortion by noise contamination for robust MFCC feature extraction, leading to uniform segmental statistics (Viikki & Laurila 1998). The individual branches were trained via Binary Cross Entropy loss. Further, the contrastive loss as formulated in Eq. (5.4.1), was applied to enforce the audio-visual consistency. Segment-wise visual and audio features from the 10th fully connected layer of each network were extracted to compute Contrastive Loss, therefore imposing the high-level audio and visual stream feature representations to be closer for the videos of the original class than those of the manipulated class.

$$L_c = \frac{1}{N}\sum_{i=1}^{N-1} \quad y^i log({d_t}^i)^2 + (1 - y^i) max(margin - {d_t}^i, 0)^2$$

(5.4.1)

Where ${d_t}^i$ corresponds to the 2-norm of dense features extracted from the fully connected layer. Parameter *margin* was set to 5.

Evaluation protocol

Quantitative evaluation of the trained CNN models was conducted by first computing the dissimilarity score ${d_t}^i$ (cf. Eq. (5.4.1)) for each video segment $t \in [1,10]$ and averaging to derive an overall prediction score per test sample. The score was compared against an empirically estimated decision threshold, the value of which was computed by the class-conditional MDS distributions in the training set and was found to to be $\tau = 0.17$. Performance was measured based on balanced accuracy and AUC.

5.4.4. Results and outlook

Table 5.4.1 presents the **performance of our pipeline for two different training settings**. In the first training setting, model updates depend only on each branch's cross-entropy loss, while in the second the contrastive loss is added resulting in much higher performance levels.

	Balanced acc. / AUC
Multimodal binary cross-entropy only	0.804 / 0.800
Multimodal binary cross-entropy & contrastive loss	0.862 / 0.860

Table 5.4.1. Performance on DFDC dataset with and without the contrastive loss.

Further experiments were conducted on **Fake AV Celeb dataset.** Forged and original videos were manually annotated for both visual and audio streams, providing a higher quality supervision signal for training in comparison to DFDC, where audio stream targets are assumed to coincide with video stream annotations. For audio feature extraction, (1) MFCC features were considered, followed by (2) deep representations of xlsr-2b model pre-trained on 960h of **LibriSpeech dataset** under the wav2vec2 objective and fine-tuned for ASR, and finally, (3) HuBERT pretrained model via self-supervision was used for feature extraction in series with RawNet3 for classification (Li et al. 2023). Training was conducted for 100 epochs with batch size 9, using two NVIDIA GeForce RTX 4080 with data and model parallel distributed training. Learning rate was initialized to 0.001 and weight regularization with L2 loss (lambda=0.0005) was also considered. The evaluation was conducted in closed-set conditions for 30% of original FakeAVCeleb dataset videos, and the evaluation outcomes are presented in Table 5.4.2.

Audio feature extractor	AUC
MFCC	0.762
Wav2vec2	0.802
HuBERT (+RawNet3)	0.781

Table 5.4.2. Comparison of different audio feature extraction methods. Performance is measured on the FakeAVCeleb dataset. Frame-level AUC is reported, by assuming a singular label per video, for all frames.

In the next period, we plan to investigate methodological improvements on the fake traces extraction mechanism that will result in improved multimodal representations, and set the basis for a more robust solution with good generalization capabilities.

5.4.5. Exposed API for integration

Currently, **we have not integrated the presented multimodal deepfake detection model in an API** due to performance considerations. Once a mature model is available, it will be integrated in the existing deepfake detection API that is described in section 4.2.5.

6. Disinformation Warning System

6.1. Introduction

As part of T3.4, we are working on developing the **Disinformation Warning System** technology (hereinafter also referred to as DWS.) This effort is guided by GDI and supported by all the remaining technology-providing partners of WP3. As defined in the AI4TRUST Grant Agreement, **the DWS system *"will label content as verified or manipulated explaining the motivation of the choices so that they are understandable by media professionals."*** In fact, the DWS can provide information on which features were most highly weighted by the model as motivation for which scores are high and low. The DWS is going to integrate outputs from certain technologies developed in tasks T3.1, T3.2 and T3.3. Additionally, GDI's data platform will be leveraged as a feature within the DWS. It will display for end users an assessment stating whether a piece of content is likely to contain disinformation or not, with a confidence score. As discussed with WP3 and WP5 partners, the DWS will be integrated into the AI4TRUST platform in the custom analysis section. Further information on the integration of the DWS in the AI4TRUST platform is available in the section below.

6.2. Data platform

The DWS will leverage, as an indicator among others, GDI's data platform to help determine if a piece of content is at risk of including disinformation or not. This data platform is indexed on GDI's definition of disinformation. GDI operates with a unique definition of disinformation which is a framework that allows for a broader range of categories and avoids the typical true versus false problems that occur when making determinations about disinformation. Specifically, GDI views disinformation through the lens of **adversarial narrative**[50] conflict. GDI defines disinformation as the intentional promotion of a misleading narrative, shared in an implicit or explicit way, that is adversarial in nature against an at-risk individual/groups[51] or institution (such as a scientific consensus including climate change or democratically-elected governments), and most importantly, creates a risk of harm. This framework has been particularly useful in GDI's work to move beyond the true vs false dichotomy. Malicious actors often use a mix of cherry pick facts to design disinformation campaigns. Some of the most pernicious and effective disinformation, which

[50] https://www.disinformationindex.org/blog/2019-8-1-adversarial-narratives-are-the-new-model-for-disinformation/

[51] GDI references the UN framework on at-risk individuals or groups, see here for more information.

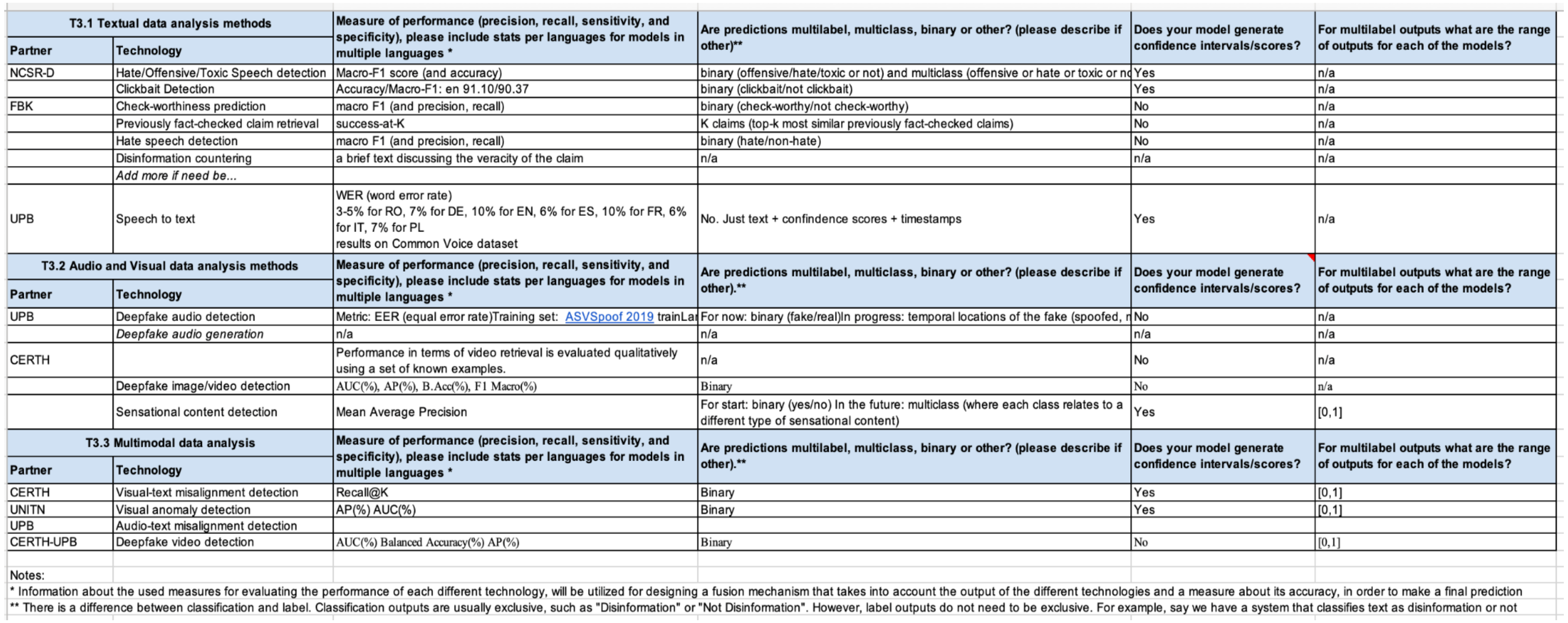
is relatable to mainstream audiences, tends to include at some level factual elements which are presented in a distorted manner to lead its audience to misleading conclusions.

In the DWS, GDI will include, as one of the features in the model, **a list of domains in the relevant languages of the AI4TRUST project**, which have been labeled by the GDI team as spreading disinformation.[52] The following sections provide further information on the interaction between GDI's data platform, and other features tied to the integration of technologies from other tasks in WP3.

6.3 Technology integration

On top of the GDI's data platform, the DWS will integrate outputs of certain technologies developed in tasks T3.1-T3.3. In the kick-off meeting of the DWS task, at the second in-person meeting of AI4TRUST in Thessaloniki Greece (M9; 25-26 September 2023), WP3 partners agreed to **develop a common understanding of the output signals of WP3 technologies to carry out the integration into the DWS**. After the launch of T3.4 at the second in-person meeting, GDI created a template to collect information on the different technologies developed in tasks T3.1-T3.3. This information has provided a **useful overview of the utilized input, output, and ground-truth data** by the aforementioned technologies.

[52] For more information on this process, you can find further information on GDI's website here.



T3.1 Textual data analysis methods		Measure of performance (precision, recall, sensitivity, and specificity), please include stats per languages for models in multiple languages *	Are predictions multilabel, multiclass, binary or other? (please describe if other)**	Does your model generate confidence intervals/scores?	For multilabel outputs what are the range of outputs for each of the models?
Partner	Technology				
NCSR-D	Hate/Offensive/Toxic Speech detection	Macro-F1 score (and accuracy)	binary (offensive/hate/toxic or not) and multiclass (offensive or hate or toxic or no	Yes	n/a
	Clickbait Detection	Accuracy/Macro-F1: en 91.10/90.37	binary (clickbait/not clickbait)	Yes	n/a
FBK	Check-worthiness prediction	macro F1 (and precision, recall)	binary (check-worthy/not check-worthy)	No	n/a
	Previously fact-checked claim retrieval	success-at-K	K claims (top-k most similar previously fact-checked claims)	No	n/a
	Hate speech detection	macro F1 (and precision, recall)	binary (hate/non-hate)	No	n/a
	Disinformation countering	a brief text discussing the veracity of the claim	n/a	n/a	n/a
	Add more if need be...				
UPB	Speech to text	WER (word error rate) 3-5% for RO, 7% for DE, 10% for EN, 6% for ES, 10% for FR, 6% for IT, 7% for PL results on Common Voice dataset	No. Just text + confindence scores + timestamps	Yes	n/a
T3.2 Audio and Visual data analysis methods		**Measure of performance (precision, recall, sensitivity, and specificity), please include stats per languages for models in multiple languages ***	**Are predictions multilabel, multiclass, binary or other? (please describe if other).****	**Does your model generate confidence intervals/scores?**	**For multilabel outputs what are the range of outputs for each of the models?**
Partner	**Technology**				
UPB	Deepfake audio detection	Metric: EER (equal error rate)Training set: ASVSpoof 2019 trainLa	For now: binary (fake/real)In progress: temporal locations of the fake (spoofed, n	No	n/a
	Deepfake audio generation	n/a	n/a	n/a	n/a
CERTH		Performance in terms of video retrieval is evaluated qualitatively using a set of known examples.	n/a	No	n/a
	Deepfake image/video detection	AUC(%), AP(%), B.Acc(%), F1 Macro(%)	Binary	No	n/a
	Sensational content detection	Mean Average Precision	For start: binary (yes/no) In the future: multiclass (where each class relates to a different type of sensational content)	Yes	[0,1]
T3.3 Multimodal data analysis		**Measure of performance (precision, recall, sensitivity, and specificity), please include stats per languages for models in multiple languages ***	**Are predictions multilabel, multiclass, binary or other? (please describe if other).****	**Does your model generate confidence intervals/scores?**	**For multilabel outputs what are the range of outputs for each of the models?**
Partner	**Technology**				
CERTH	Visual-text misalignment detection	Recall@K	Binary	Yes	[0,1]
UNITN	Visual anomaly detection	AP(%) AUC(%)	Binary	Yes	[0,1]
UPB	Audio-text misalignment detection				
CERTH-UPB	Deepfake video detection	AUC(%) Balanced Accuracy(%) AP(%)	Binary	No	[0,1]

Notes:
* Information about the used measures for evaluating the performance of each different technology, will be utilized for designing a fusion mechanism that takes into account the output of the different technologies and a measure about its accuracy, in order to make a final prediction
** There is a difference between classification and label. Classification outputs are usually exclusive, such as "Disinformation" or "Not Disinformation". However, label outputs do not need to be exclusive. For example, say we have a system that classifies text as disinformation or not

Figure 6.1. Screenshot of the created and circulated template to streamline the collection of information from WP3 partners.

For the third in-person meeting of the AI4TRUST project held at the CNRS premises in Paris on 8 and 9 February 2024, GDI elaborated an initial **work plan for the integration of the developed data analysis technologies in the DWS** (see Tasks T3.1-T3.3 of WP3). Upon the presentation of this work plan and based on the fact that the DWS will be used to flag large collections of content ingested in the AI4TRUST platform through the social media listening module, **WP3 partners agreed that only a subset of these technologies should be taken into account for integration into the DWS**; namely, only the technologies that can carry out large scale data analysis. The selected technologies are listed in Table 6.1; though, this list can be updated and extended throughout the life of the project, depending on evolution of the technologies developed in T3.1-T3.3. Table 6.1 indicates the preliminary selection of technologies which will be integrated in the first pilot of the DWS:

WP task	Patner	Technology
T3.1	NCSR-D	Hate Speech detection
T3.1	NCSR-D	Offensive language detection
T3.1	NCSR-D	Clickbait Detection
T3.1	FBK	Checkworthy Claim Detection
T3.2	CERTH	Sensational content detection
T3.3	CERTH	Visual-text misalignment detection
T3.3	UNITN	Visual anomaly detection

Table 6.1: List of technologies that will be integrated in the first version of the DWS.

With respect to the integration of the above listed technologies, GDI presented an initial framework that is depicted in figure 6.2 below. The current work plan is to develop an ensemble classifier, which will embed the outputs of the selected technologies (see Table 6.1) from T3.1-T3.3 into a new statistical model. The outputs of these technologies will serve as features in this model (in a similar capacity to GDI's data platform). The current design of the DWS will display a first level of classification based on the format of the online content filtered through the tool (image, text, video). The model will use these scores to carry out a final weighting. The **overall output** of this model will be a **score indicating if a piece of content is likely to contain disinformation/not**.

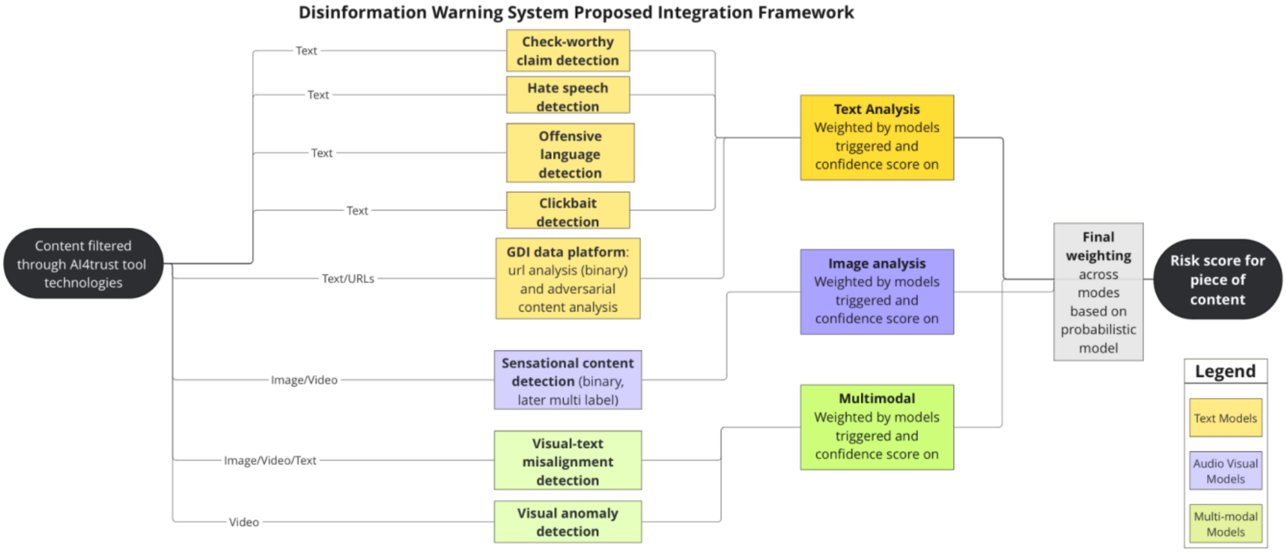

Disinformation Warning System Proposed Integration Framework
Content filtered through AI4trust tool technologies
Text
Text
Text
Text
Text/URLs
Image/Video
Image/Video/Text
Video
Check-worthy claim detection
Hate speech detection
Offensive language detection
Clickbait detection
GDI data platform: url analysis (binary) and adversarial content analysis
Sensational content detection (binary, later multi label)
Visual-text misalignment detection
Visual anomaly detection
Text Analysis Weighted by models triggered and confidence score on
Image analysis Weighted by models triggered and confidence score on
Multimodal Weighted by models triggered and confidence score on
Final weighting across modes based on probabilistic model
Risk score for piece of content
Legend
Text Models
Audio Visual Models
Multi-modal Models

Figure 6.2: The current framework for the integration of the Disinformation Warning System. This framework will be adapted throughout the project depending on the added functionalities of different technologies of WP3 and languages.

The integration of the DWS in the AI4Trust tool. The integration of the Disinformation Warning System into the over AI4TRUST tool is an ongoing technical discussion between WP3 and WP5 partners, further details will be provided in the next WP3 and WP5 deliverables (i.e., D3.1 and D5.5).

6.4. Warning mechanism

The DWS will display for end users if a piece of content is likely to contain disinformation or not. The model will output a probability score between 0 and 1, with 1 being highly likely to be disinformation. At this stage, GDI is envisioning score brackets (e.g. 0.8 -1 is high risk, 0-0.2 is low risk) which would indicate if a piece of content has a high/ average/low risk of disinformation. Also, GDI suggests the use of a green/amber/red system to represent the likelihood of the risk of a piece of content being disinformation/not based on the probability score outputted by the DWS. This specific point of the warning mechanism needs to be further discussed with consortium partners. Additionally, a **confidence score** will be displayed besides the probability score for each piece of content, as an additional indicator for the end user.

6.5. Current progress and outlook

At the moment, **GDI is in the process of collecting additional information from WP3 partners** who are developing technologies which will be integrated into the DWS. After the completion of this process, GDI will start working on releasing a first version of the DWS.

7. Conclusions and next steps

In this deliverable, we presented the **first version of the developed technologies in WP3** and provided details about the exposed APIs for the integration of some of them in the AI4TRUST platform. We started by describing a set of text analysis methods for detecting disinformation signals, assessing the check-worthiness of a given claim, retrieving relevant already fact-checked claims, and generating a verdict about the veracity/fakeness of a claim. Afterwars, we presented the audio analysis tools for speech-to-text transcription (that can be seen as a pre-processing step of the aforementioned text analysis methods), deepfake audio detection and deepfake audio generation. Then, we reported on the released visual analysis technologies for reverse video search on the Web, deepfake image/video detection, sensational content detection and synthetic image/video generation. Finally, we described the set of multimodal analysis methods for video and audio anomaly detection, visual-text misalignment detection, and multimodal video deepfake detection, and discussed the efforts made towards building a Disinformation Warning System.

Building on the aforementioned developments, over the next months of the project we will:

- **Extend our research on text analysis technologies**, by:
  - expanding our study on disinformation signal detection in text, by developing multilingual models for argumentation mining and fact-checking;
  - extending the developed check-worthy claim detection method to support the Spanish language, and experiment with the use of balanced class weights to better deal with label imbalance and perform additional tuning;
  - extending our method for fact-checked claim retrieval to support additional languages (Spanish, German and French), and designing novel negative sampling strategies to improve the retrieval performance;
  - further investigating the impact of having multiple documents as input for verdict generation, and extending the service to support all the languages of the project.
- **Continue our work on audio analysis and generation technologies**, by:
  - using fast conformer transducer models for languages besides Romanian, and creating new models for additional languages envisaged by the project such as Italian, German, and French;
  - evaluating the audio deepfake detection method in a real-world scenario using real and fake audio data distributed in social media;
  - targeting the generation of fake samples in multiple languages, and trying to combine all the trained text-to-mel models with various neural and signal-based vocoders.

- **Advance our visual analysis and generation technologies**, by:
    - automating the interaction of the tool for reverse video search on the Web with additional search engines, providing information about the publication date of the retrieved videos, and allowing the detection of near-duplicates of the query video, also in closed collections;
    - performing a systematic evaluation of different options for deepfake image/video detection (including more recent state of the art methods on a wide range of datasets and evaluation settings), as well as investigating novel contributions on top of the best performing methodologies;
    - exploring the use of Large Multimodal Model (LMM) based knowledge in order to improve the performance of sensational content detection;
    - exploring more sophisticated designs for synthetic image/video generation, while maintaining the core object-level formulation; i.e. we plan to extend the explicit control over other aspects of the objects, (e.g., illumination, pose, etc.), and improve the identity preservation of the edited object.
- **Expand our developments on multimodal data analysis**, by:
    - extending the existing approach for video anomaly detection to automatically identify and classify anomalous events or behaviors in videos without relying on predefined categories or labels;
    - validat ing our audio anomaly detection approach on challenging out-of-domain datasets (e.g., LAV-DF or HalfTruth);
    - experimenting with additional datasets and approaches for generating training data, and exploring the use of different backbone models and training approaches to further improve the visual-text misalignment detection performance;
    - investigating methodological improvements on the fake traces extraction mechanism of the developed multimodal approach for deepfake video detection, that will result in improved multimodal representations and set the basis for a more robust solution with good generalization capabilities.
- **Work towards integrating the selected data analysis technologies into the Disinformation Warning System (DWS)**, and releasing the first version of this tool.

8. References

Ahmadi, N., Lee, J., Papotti, P., & Saeed, M. (2019). Explainable fact checking with probabilistic answer set programming. arXiv preprint arXiv:1906.09198.

Alhindi, T., Petridis, S., & Muresan, S. (2018, November). Where is your evidence: Improving fact-checking by justification modeling. In Proceedings of the first workshop on fact extraction and verification (FEVER).

Alhindi, T., McManus, B., & Muresan, S. (2021). What to Fact-Check: Guiding Check-Worthy Information Detection in News Articles through Argumentative Discourse Structure. In Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, Singapore and Online (pp. 380–391). Association for Computational Linguistics.

Aneja, S., Bregler, C., & Nießner, M. (2021). Cosmos: Catching out-of-context misinformation with self-supervised learning. arXiv preprint arXiv:2101.06278.

Antoun, W., Baly, F., & Hajj, H. (2020). AraBERT: Transformer-based Model for Arabic Language Understanding. In Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, Marseille, France (pp. 9-15). European Language Resource Association.

Ap-Apid, R. (2005, March). An algorithm for nudity detection. In 5th Philippine Computing Science Congress (pp. 201-205).

Apostolidis, E., Balaouras, G., Mezaris,V., & Patras, I. (2023). Selecting a Diverse Set of Aesthetically-pleasing and Representative Video Thumbnails using Reinforcement Learning. In Proceedings of the IEEE Int. Conf. on Image Processing (ICIP 2023), Kuala Lumpur, Malaysia, Oct. 2023.

Atanasova, P., Simonsen, J. G., Lioma, C., & Augenstein, I. (2020). Generating fact checking explanations. arXiv preprint arXiv:2004.05773.

Avrahami, O., et al. SpaText: Spatio-Textual Representation for Controllable Image Generation, arXiv:2211.14305, 2022.

Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. In *Proc. NeurIPS*.

Bai, S., He, Z., Lei, Y., Wu, W., Zhu, C., Sun, M., Yan, J., (2019). Traffic anomaly detection via perspective map based on spatial-temporal information matrix. in CVPR Workshops.

Barrón-Cedeño, A., Alam, F., Galassi, A., Da San Martino, G., Nakov, P., Elsayed, T., Azizov, D., Caselli, T., Cheema, G. S., Haouari, F., Hasanain, M., Kutlu, M., Li, C., Ruggeri, F., Struß, J. M., & Zaghouani W. (2023). Overview of the CLEF–2023 CheckThat! Lab on Checkworthiness, Subjectivity, Political Bias, Factuality, and Authority of News Articles and Their Source. In Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2023. Lecture Notes in Computer Science, 14163. Springer, Cham.

Barrón-Cedeño, A., Elsayed, T., Nakov, P., Da San Martino, G., Hasanain, M., Suwaileh, R., Haouari, F., Babulkov, N., Hamdan, B., Nikolov, A., Shaar, S. & Ali, Z. S. (2020). Overview of CheckThat! 2020: Automatic Identification and Verification of Claims in Social Media. In Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2020. Lecture Notes in Computer Science, 12260. Springer, Cham.

Baevski, Alexei, et al. "wav2vec 2.0: A framework for self-supervised learning of speech representations." Advances in neural information processing systems 33 (2020): 12449-12460.

Basile, Valerio & Bosco, Cristina & Fersini, Elisabetta & Nozza, Debora & Patti, Viviana & Rangel Pardo, Francisco & Rosso, Paolo & Sanguinetti, Manuela. (2019). SemEval-2019 Task 5: Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter. 54-63. 10.18653/v1/S19-2007.

Baxevanakis, S., Kordopatis-Zilos, G., Galopoulos, P., Apostolidis, L., Levacher, K., Baris Schlicht, I., ... & Papadopoulos, S. (2022, June). The mever deepfake detection service: Lessons learnt from developing and deploying in the wild. In Proceedings of the 1st International Workshop on Multimedia AI against Disinformation (pp. 59-68).

Bhatt, Umang, et al. "Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty." In *Proc. ACM Conference on AI, Ethics, and Society*. 2021.

Błasiok, J., Gopalan, P., Hu, L., & Nakkiran, P. (2024). When Does Optimizing a Proper Loss Yield Calibration? In *Proc. NeurIPS*.

Brooks, T., et al. In structpix2pix: Learning to follow image editing instructions. arXiv:2211.09800, 2022.

Cai, Z., Ghosh, S., Adatia, A. P., Hayat, M., Dhall, A., & Stefanov, K. (2023). AV-Deepfake1M: A Large-Scale LLM-Driven Audio-Visual Deepfake Dataset. arXiv preprint arXiv:2311.15308.

Cai, Z., Stefanov, K., Dhall, A., & Hayat, M. (2022, November). Do you really mean that? Content driven audio-visual deepfake dataset and multimodal method for temporal forgery localization. In 2022 International Conference on Digital Image Computing: Techniques and Applications (DICTA) (pp. 1-10). IEEE.

Cao, M., et al. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. arXiv:2304.08465, 2023

Cao, Q., Shen, L., Xie, W., Parkhi, O. M., & Zisserman, A. (2018, May). Vggface2: A dataset for recognising faces across pose and age. In 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018) (pp. 67-74). IEEE.

Capoferri, D., Borrelli, C., Bestagini, P., Antonacci, F., Sarti, A., & Tubaro, S. (2020). Speech Audio Splicing Detection and Localization Exploiting Reverberation Cues. 2020 IEEE International Workshop on Information Forensics and Security (WIFS), 1-6. https://doi.org/10.1109/WIFS49906.2020.9360900.

Carrell, A. M., Mallinar, N., Lucas, J., & Nakkiran, P. (2022). The calibration generalization gap. *arXiv preprint arXiv:2210.01964*.

Charitidis, P., Kordopatis-Zilos, G., Papadopoulos, S., & Kompatsiaris, I. (2020). Investigating the impact of pre-processing and prediction aggregation on the deepfake detection task. arXiv preprint arXiv:2006.07084.

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., ... & Wei, F. (2022a). WavLM: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, *16*(6), 1505-1518.

Chen, Z., Wu, Y., Leng, Y., Chen, J., Liu, H., Tan, X., ... & Mandic, D. (2022b). Resgrad: Residual denoising diffusion probabilistic models for text to speech. arXiv preprint arXiv:2212.14518.

Chi, Z., Dong, L., Zheng, B., Huang, S., Mao, X.-L., Huang, H., & Wei, F. (2021, August). Improving Pretrained Cross-Lingual Language Models via Self-Labeled Word Alignment. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (pp. 3418–3430). doi:10.18653/v1/2021.acl-long.265

Chi, Z., Dong, L., Wei, F., Yang, N., Singhal, S., Wang, W., ... Zhou, M. (2021). InfoXLM: An Information-Theoretic Framework for Cross-Lingual Language Model Pre-Training. arXiv [Cs.CL]. Retrieved from http://arxiv.org/abs/2007.07834

Choi, S., et al. RobustNet: Improving domain generalization in urban-scene segmentation via instance selective whitening. CVPR 2021.

Chugh, K., Gupta, P., Dhall, A., & Subramanian, R. (2020, October). Not made for each other-audio-visual dissonance-based deepfake detection and localization. In Proceedings of the 28th ACM international conference on multimedia (pp. 439-447).

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., ... Wei, J. (2022). Scaling Instruction-Finetuned Language Models. arXiv [Cs.LG]. Retrieved from http://arxiv.org/abs/2210.11416

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., ... Stoyanov, V. (2020). Unsupervised Cross-lingual Representation Learning at Scale. arXiv [Cs.CL]. Retrieved from http://arxiv.org/abs/1911.02116

Conneau, A., Baevski, A., Collobert, R., Mohamed, A., & Auli, M. (2021). Unsupervised cross-lingual representation learning for speech recognition. In *Proc. Interspeech*.

Cordts, M., et al. The cityscapes dataset for semantic urban scene understanding. CVPR 2016.

Couairon, G., et al. DiffEdit: Diffusion-based semantic image editing with mask guidance, arXiv:2210.11427, 2022.

Dai, Q., Zhao, R. W., Wu, Z., Wang, X., Gu, Z., Wu, W., & Jiang, Y. G. (2015, September). Fudan-Huawei at MediaEval 2015: Detecting Violent Scenes and Affective Impact in Movies with Deep Learning. In MediaEval (Vol. 1436).

Damstra, A., Boomgaarden, H. G., Broda, E., Lindgren, E., Stromback, J., Tsfati, Y., & Vliegenthart, R. (2021). What does fake look like? A review of the literature on intentional deception in the news and on social media. Journalism Studies, 22(14), 1947–1963.

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. Proceedings of the International AAAI Conference on Web and Social Media, 11(1), 512-515. https://doi.org/10.1609/icwsm.v11i1.14955

Deng, J., et al. ImageNet: A large-scale hierarchical image database. CVPR 2009.

Devlin, J., Chang, M-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Minneapolis, Minnesota (pp. 4171–4186). Association for Computational Linguistics.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., ... & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

Dolhansky, B., Bitton, J., Pflaum, B., Lu, J., Howes, R., Wang, M., & Ferrer, C. C. (2020). The deepfake detection challenge (dfdc) dataset. arXiv preprint arXiv:2006.07397.

Du, S. K. N. M. & Gollapalli, S. D. (2022). NUS-IDS at CheckThat! 2022: Identifying check-worthiness of tweets using CheckthaT5. In Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, CLEF '2022, Bologna, Italy.

Elsayed, T., Nakov, P., Barrón-Cedeño, A., Hasanain, M., Suwaileh, R., Da San Martino, G., & Atanasova, P. (2019). Overview of the CLEF-2019 CheckThat! Lab: Automatic Identification and Verification of Claims. In Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2019. Lecture Notes in Computer Science, 11696. Springer, Cham.

Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., ... Joulin, A. (2020). Beyond English-Centric Multilingual Machine Translation. arXiv [Cs.CL]. Retrieved from http://arxiv.org/abs/2010.11125

Fröbe, M., Stein, B., Gollub, T., Hagen, M., & Potthast, M. (2023). SemEval-2023 Task 5: Clickbait Spoiling. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2275–2286, Toronto, Canada. Association for Computational Linguistics.

Fu, Zhikang, et al. "PornNet: a unified deep architecture for pornographic video recognition." Applied Sciences 11.7 (2021): 3066.

Gangwar, A., González-Castro, V., Alegre, E., & Fidalgo, E. (2021). AttM-CNN: Attention and metric learning based CNN for pornography, age and Child Sexual Abuse (CSA) Detection in images. Neurocomputing, 445, 81-104.

Gad-Elrab, M. H., Stepanova, D., Urbani, J., & Weikum, G. (2019, January). Exfakt: A framework for explaining facts over knowledge graphs and text. In Proceedings of the twelfth ACM international conference on web search and data mining (pp. 87-95).

Georgescu, A. L., Cucu, H., & Burileanu, C. (2021, October). Improvements of speed's romanian asr system during reterom project. In 2021 International Conference on Speech Technology and Human-Computer Dialogue (SpeD) (pp. 177-182). IEEE.

Giannakopoulos, T., Kosmopoulos, D., Aristidou, A., & Theodoridis, S. (2006). Violence content classification using audio features. In Advances in Artificial Intelligence: 4th Helenic Conference on AI, SETN 2006, Heraklion, Crete, Greece, May 18-20, 2006. Proceedings 4 (pp. 502-507). Springer Berlin Heidelberg.

Goel, V., et al, PAIR Diffusion: A Comprehensive Multimodal Object-Level Image Editor, CVPR 2024.

Gulati, Anmol, et al. "Conformer: Convolution-augmented transformer for speech recognition." arXiv preprint arXiv:2005.08100 (2020).

Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In *Proc. ICML* (pp. 1321-1330).

Guo, Z., Schlichtkrull, M., & Vlachos, A. (2022). A Survey on Automated Fact-Checking. Transactions of the Association for Computational Linguistics, 10, 178-206.

Gupta, A., Li, H., Farnoush, A., & Jiang, W. (2022). Understanding patterns of COVID infodemic: A systematic and pragmatic approach to curb fake news. Journal of Business Research, 140(February), 670–683.

Hagen, M., Fröbe, M., Jurk, A., & Potthast, M. (2022). Webis Clickbait Spoiling Corpus 2022 (1.0.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.6362726

Hamby, A., Kim, H., & Spezzano, F. (2024). Sensational stories: The role of narrative characteristics in distinguishing real and fake news and predicting their spread. Journal of

Business Research, Volume 170, 2024, 114289, ISSN 0148-2963

Hara, K., Kataoka, H., & Satoh, Y. (2018). Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (pp. 6546-6555).

Hardalov, M., Chernyavskiy, A., Koychev, I., Ilvovsky, D., & Nakov, P. (2022). CrowdChecked: Detecting Previously Fact-Checked Claims in Social Media. In Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online only (pp. 266–285). Association for Computational Linguistics.

He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

He, P., Liu, X., Gao, J., & Chen, W. (2021). DeBERTa: Decoding-enhanced BERT with Disentangled Attention. arXiv [Cs.CL]. Retrieved from http://arxiv.org/abs/2006.03654

Hertz, A., et al. Prompt-to-prompt image editing with cross attention control. arXiv:2208.01626, 2022.

Ho, J., Kalchbrenner, N., Weissenborn, D., Salimans, T. (2019). Axial attention in multidimensional transformers. arXiv.

Hu, M., et al. Unified Discrete Diffusion for Simultaneous Vision-Language Generation, arXiv:2211.14842, 2022.

Huang, J., et al. FSDR: Frequency space domain randomization for domain generalization. CVPR 2021.

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. *arXiv preprint arXiv:1602.07360*.

Ireton, C., & Posetti, J. (2018). Journalism, Fake News & Disinformation: Handbook for Journalism Education and Training. Unesco Publishing.

Jadhav, S., Patole, R., & Rege, P. (2019). Audio Splicing Detection using Convolutional Neural Network. 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 1-5. https://doi.org/10.1109/icccnt45670.2019.8944345.

Ju, Z., Wang, Y., Shen, K., Tan, X., Xin, D., Yang, D., ... & Zhao, S. (2024). NaturalSpeech 3: Zero-Shot Speech Synthesis with Factorized Codec and Diffusion Models. arXiv preprint arXiv:2403.03100.

Kazemi, A., Garimella, K., Gaffney, D., & Hale, S. (2021). Claim Matching Beyond English to Scale Global Fact-Checking. In Proceedings of the 59th Annual Meeting of the Association for

Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online (pp. 4504–4517). Association for Computational Linguistics.

Khalid, H., Tariq, S., Kim, M., & Woo, S. S. (2021). FakeAVCeleb: A novel audio-video multimodal deepfake dataset. arXiv preprint arXiv:2108.05080.

Kim, Sehoon, et al. "Squeezeformer: An efficient transformer for automatic speech recognition." Advances in Neural Information Processing Systems 35 (2022): 9361-9373.

Kim, S., Kim, H., & Yoon, S. (2022). Guided-TTS 2: A diffusion model for high-quality adaptive text-to-speech with untranscribed data. arXiv preprint arXiv:2205.15370.

Kim, H., Kim, S., Yeom, J., & Yoon, S. (2023). UnitSpeech: Speaker-adaptive speech synthesis with untranscribed data. Proceedings of Interspeech, Dublin, Ireland, 2023

Danijel Koržinek, Krzysztof Marasek, Łukasz Brocki, and Krzysztof Wołk Polish read speech corpus for speech tools and services In Selected papers from the CLARIN Annual Conference 2016, Aix-en-Provence, 26–28 October 2016, CLARIN Common Language Resources and Technology Infrastructure, number 136, pages 54–62. Linköping University Electronic Press, Linköpings universitet, 2017.

Koonce, B., Koonce, B. (2021). EfficientNet. *Convolutional neural networks with swift for Tensorflow: image recognition and dataset categorization*, 109-123.

Kotonya, N., & Toni, F. (2020, December). Explainable Automated Fact-Checking: A Survey. In Proceedings of the 28th International Conference on Computational Linguistics (pp. 5430-5443)

Kotonya, N., & Toni, F. (2020, November). Explainable Automated Fact-Checking for Public Health Claims. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 7740-7754).

Kuchaiev, Oleksii, et al. "Nemo: a toolkit for building ai applications using neural modules." arXiv preprint arXiv:1909.09577 (2019).

Kumar, A., Bhavsar, A., & Verma, R. (2020, April). Detecting deepfakes with metric learning. In 2020 8th international workshop on biometrics and forensics (IWBF) (pp. 1-6). IEEE.

Kumar, N., Narang, A., & Lall, B. (2022). Zero-Shot Normalization Driven Multi-Speaker Text to Speech Synthesis. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 30, 1679-1693. https://doi.org/10.1109/taslp.2022.3169634.

Tiedemann, J., & Thottingal, S. (2020). OPUS-MT – Building open translation services for the World. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Łańcucki, A. (2021, June). Fastpitch: Parallel text-to-speech with pitch prediction. In ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6588-6592). IEEE.
Lample, G., & Conneau, A. (2019). Cross-lingual Language Model Pretraining. arXiv [Cs.CL]. Retrieved from http://arxiv.org/abs/1901.07291
Larocque, W. (2021). Gore classification and censoring in images (Doctoral dissertation, Université d'Ottawa/University of Ottawa).
Li, Y., Yang, X., Sun, P., Qi, H., & Lyu, S. (2020). Celeb-df: A large-scale challenging dataset for deepfake forensics. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 3207-3216).
Li, L., Lu, T., Ma, X., Yuan, M., & Wan, D. (2023). Voice Deepfake Detection Using the Self-Supervised Pre-Training Model HuBERT. Applied Sciences, 13(14), 8488.
Li, G., Cai, G., Zeng, X., Zhao, R., (2022a). Scale-aware spatio-temporal relation learning for video anomaly detection. ECCV.
Li, S., Liu, F., Jiao, L., (2022b). Self-training multi-sequence learning with transformer for weakly supervised video anomaly detection. AAAI.
Li, Y., & Xie, Y. (2020). Is a picture worth a thousand words? An empirical study of image content and social media engagement. Journal of marketing research, 57(1), 1-19.
Liew, J., et al. MagicMix: Semantic Mixing with Diffusion Models, arXiv:2210.16056, 2022.
Lin, C. Y. (2004, July). Rouge: A package for automatic evaluation of summaries. In Text summarization branches out (pp. 74-81).
Liu, N., et al. (2022). Compositional Visual Generation with Composable Diffusion Models, arXiv:2206.01714, 2022.
Liu, F., Wang, Y., Wang, T., & Ordonez, V. (2020). Visual news: Benchmark and challenges in news image captioning. arXiv preprint arXiv:2010.03743.
Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.
Liu, S., Su, D., & Yu, D. (2022). Diffgan-tts: High-fidelity and efficient text-to-speech with denoising diffusion gans. arXiv preprint arXiv:2201.11972.
Liu, H., Dai, Z., So, D., & Le, Q. V. (2021). Pay attention to MLPs. In *Proc. NeurIPS*.
Liu, H., Chen, Z., Yuan, Y., Mei, X., Liu, X., Mandic, D., & Plumbley, M. D. (2023). AudioLDM: Text-

to-audio generation with latent diffusion models. In International conference on machine learning. PMLR.

Liu, Z., Guo, Y., & Yu, K. (2023b). DiffVoice: Text-to-Speech with Latent Diffusion. In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1-5). IEEE.

Liu, W., Luo, W., Lian, D., Gao, S. (2018). Future frame prediction for anomaly detection–a new baseline. CVPR

Liu, Z., Nie, Y., Long, C., Zhang, Q., Li, G., (2021). A hybrid video anomaly detection framework via memory-augmented flow reconstruction and flowguided frame prediction. ICCV

Lopes, A. P., de Avila, S. E., Peixoto, A. N., Oliveira, R. S., & Araújo, A. D. A. (2009, August). A bag-of-features approach based on hue-sift descriptor for nude detection. In 2009 17th European Signal Processing Conference (pp. 1552-1556).

Lu, Y. J., & Li, C. T. (2020, July). GCAN: Graph-aware Co-Attention Networks for Explainable Fake News Detection on Social Media. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (pp. 505-514).

Luo, G., Darrell, T., & Rohrbach, A. (2021). Newsclippings: Automatic generation of out-of-context multimodal media. arXiv preprint arXiv:2104.05893.

Lv, H., Chen, C., Cui, Z., Xu, C., Li, Y., Yang, J., (2021). Learning normal dynamics in videos with meta prototype network. CVPR.

Masood, M., Nawaz, M., Malik, K. M., Javed, A., Irtaza, A., & Malik, H. (2023). Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward. *Applied intelligence*, *53*(4), 3974-4026.

Matern, F., Riess, C., & Stamminger, M. (2019, January). Exploiting visual artifacts to expose deepfakes and face manipulations. In 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW) (pp. 83-92). IEEE.

Mehta, S., Tu, R., Beskow, J., Székely, É., & Henter, G. E. (2023). Matcha-TTS: A fast TTS architecture with conditional flow matching. arXiv preprint arXiv:2309.03199.

Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010, September). Recurrent neural network based language model. In Interspeech (Vol. 2, No. 3, pp. 1045-1048)

Moussa, Denise, et al. "Point to the Hidden: Exposing Speech Audio Splicing via Signal Pointer Nets." arXiv preprint arXiv:2307.05641 (2023).

Moustafa, M. (2015). Applying deep learning to classify pornographic images and videos. arXiv preprint arXiv:1511.08899.

Mridha, M. F., Keya, A. J., Hamid, M. A., Monowar, M. M., & Rahman, M. S. (2021). A comprehensive review on fake news detection with deep learning. IEEE access, 9, 156151-156170.

Mu, G., Cao, H., & Jin, Q. (2016). Violent scene detection using convolutional neural networks and deep audio features. In Pattern Recognition: 7th Chinese Conference, CCPR 2016, Chengdu, China, November 5-7, 2016, Proceedings, Part II 7 (pp. 451-463). Springer Singapore.

Müller, N. M., Czempin, P., Dieckmann, F., Froghyar, A., & Böttinger, K. (2022). Does audio deepfake detection generalize? In *Proc. Interspeech*.

Nadeem, M. S. A., Zucker, J. D., & Hanczar, B. (2009, March). Accuracy-rejection curves (ARCs) for comparing classification methods with a reject option. In *Machine Learning in Systems Biology* (pp. 65-81). PMLR.

Nakamura, K., Levy, S., Wang, W.Y. (2020) Fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection. In: Proceedings of the twelfth language resources and evaluation conference, pp 6149–6157

Nakov, P., Barrón-Cedeño, A., Da San Martino, G., Alam, F., Míguez, R., Caselli, T., Kutlu, M., Zaghouani, W., Li, C., Shaar, S. and Mubarak, H. (2022). Overview of the CLEF-2022 CheckThat! lab task 1 on identifying relevant claims in tweets. In 2022 Conference and Labs of the Evaluation Forum, CLEF 2022 (pp. 368-392). CEUR-WS.org.

Nakov, P., Da San Martino, G., Elsayed, T., Barrón-Cedeño, A., Míguez, R., Shaar, S., Alam, F., Haouari, F., Hasanain, M., Mansour, W., Hamdan, B., Ali, Z. S., Babulkov, N., Nikolov, A., Shahi, G. K., Struß, J. M., Mandl, T., Kutlu, M., & Kartal, Y. S. (2021). Overview of the CLEF–2021 CheckThat! Lab on Detecting Check-Worthy Claims, Previously Fact-Checked Claims, and Fake News. In Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2021. Lecture Notes in Computer Science, 12880. Springer, Cham.

Nakov, P., Barrón-Cedeño, A., Elsayed, T., Suwaileh, R., Màrquez, L., Zaghouani, W., Atanasova, P., Kyuchukov, S. & Da San Martino, G. (2018). Overview of the CLEF-2018 CheckThat! Lab on Automatic Identification and Verification of Political Claims. In Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2018. Lecture Notes in Computer Science, 11018. Springer, Cham.

Narasimhan, M.G., (2018). Dynamic video anomaly detection and localization using sparse denoising autoencoders. Multimedia Tools and Applications.

Neuhold, G., et al. The mapillary vistas dataset for semantic understanding of street scenes. ICCV 2017.

Newman, E. J., Garry, M., Bernstein, D. M., Kantner, J., & Lindsay, D. S. (2012). Nonprobative photographs (or words) inflate truthiness. Psychonomic Bulletin & Review, 19, 969-974.

Ni, J., Qu, C., Lu, J., Dai, Z., Abrego, G. H., Ma, J., Zhao, V., Luan, Y., Hall, K., Chang, M.-W., & Yang, Y. (2022). Large Dual Encoders Are Generalizable Retrievers. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, United Arab Emirates (pp. 9844–9855). Association for Computational Linguistics.

Nirkin, Y., Wolf, L., Keller, Y., & Hassner, T. (2021). Deepfake detection based on discrepancies between faces and their context. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(10), 6111-6121.

Ojha, Utkarsh, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *Proc. CVPR*. 2023.

Pan, X., Zhang, X., & Lyu, S. 2012. Detecting splicing in digital audios using local noise level estimation. 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1841-1844. https://doi.org/10.1109/ICASSP.2012.6288260.

Pan, X., et al. 2018. Two at once: Enhancing learning and generalization capacities via IBN-Net. ECCV 2018.

Panayotov, Vassil, et al. "Librispeech: an asr corpus based on public domain audio books." 2015 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2015.

Panchendrarajan R., & Zubiaga, A. (2024). Claim Detection for Automated Fact-checking: A Survey on Monolingual, Multilingual and Cross-Lingual Research. arXiv preprint 2401.11969.

Papadopoulos, S.I., Koutlis, C., Papadopoulos S, et al (2023). Synthetic misinformers: Generating and combating multimodal misinformation. In: Proceedings of the 2nd ACM international workshop on multimedia AI against Disinformation, pp 36–44

Papadopoulos, S. I., Koutlis, C., Papadopoulos, S., & Petrantonakis, P. C. (2024). VERITE: a Robust benchmark for multimodal misinformation detection accounting for unimodal bias. International Journal of Multimedia Information Retrieval, 13(1), 4.

Parisi, L., Francia, S., & Magnani, P. (2020). UmBERTo: an Italian language model trained with whole word masking. https://github.com/musixmatchresearch/umberto. Accessed: 2024-01-01.

Park, H., Noh, J., Ham, B., (2020). Learning memory-guided normality for anomaly detection. CVPR.

Patashnik, O., at al. Localizing object-level shape variations with text-to-image diffusion models. arXiv:2303.11306, 2023.

PBS. (2021, September 29). When sensationalism became fake news. Retrieved from https://www.pbs.org/wgbh/americanexperience/features/conversations-when-sensationalism-became-fake-news/

Peddinti, Vijayaditya, Daniel Povey, and Sanjeev Khudanpur. "A time delay neural network architecture for efficient modeling of long temporal contexts." Interspeech. 2015.

Pikuliak, M., Srba, I., Moro, R., Hromadka, T., Smoleň, T., Melišek, M., Vykopal, I., Simko, J., Podroužek, J., & Bielikova, M. (2023). Multilingual Previously Fact-Checked Claim Retrieval. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore (pp. 16477–16500). Association for Computational Linguistics.

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, *10*(3), 61-74.

Polignano, M., Basile, P., De Gemmis, M., Semeraro, G., & Basile, V. (2019). AlBERTo: Italian BERT language understanding model for NLP challenging tasks based on tweets. In 6th Italian Conference on Computational Linguistics, CLiC-it 2019. CEUR.

Popat, K., Mukherjee, S., Yates, A., & Weikum, G. (2018). Declare: Debunking fake news and false claims using evidence-aware deep learning. arXiv preprint arXiv:1809.06416.

Popov, V., Vovk, I., Gogoryan, V., Sadekova, T., & Kudinov, M. (2021, July). Grad-tts: A diffusion probabilistic model for text-to-speech. In International Conference on Machine Learning (pp. 8599-8608). PMLR. 2021

Potthast, M., Kiesel, J., Reinartz, K., Bevendorff, J., & Stein, B. (2018, July). A Stylometric Inquiry into Hyperpartisan and Fake News. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 231-240).

Povey, Daniel, et al. "The Kaldi speech recognition toolkit." IEEE 2011 workshop on automatic speech recognition and understanding. IEEE Signal Processing Society, 2011.

Povey, D., Peddinti, V., Galvez, D., Ghahremani, P., Manohar, V., Na, X., ... & Khudanpur, S. (2016, September). Purely sequence-trained neural networks for ASR based on lattice-free MMI. In Interspeech (pp. 2751-2755).

Qi, C., et al. PointNet++: Deep hierarchical feature learning on point sets in a metric space. NeurIPS 2017.

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al., (2021). Learning transferable visual models from natural language supervision. ICML

Radford, Alec, et al. "Robust speech recognition via large-scale weak supervision." International Conference on Machine Learning. PMLR, 2023.

Rana, M. S., Nobi, M. N., Murali, B., & Sung, A. H. (2022). Deepfake detection: A systematic literature review. IEEE access, 10, 25494-25513.

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese

BERT-Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China (pp. 3982–3992). Association for Computational Linguistics.

Richter, S., et al. Playing for data: Ground truth from computer games. ECCV 2016.

Robertson, S., & Zaragoza, H. (2009). The Probabilistic Relevance Framework: BM25 and Beyond. Foundations and Trends in Information Retrieval, 3(4), 333-389.

Rombach, R., et al. High-resolution image synthesis with latent diffusion models. CVPR 2022.

Ros, G., et al. The Synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. CVPR 2016.

Rossler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J., & Nießner, M. (2019). Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 1-11).

Ruiz, N., et al. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation, arXiv:2208.12242, 2022.

Russo, D., Tekiroğlu, S. S., & Guerini, M. (2023). Benchmarking the Generation of Fact Checking Explanations. Transactions of the Association for Computational Linguistics

Sabir, E., AbdAlmageed, W., Wu, Y., et al (2018). Deep multimodal image-repurposing detection. In: Proceedings of the 26th ACM international conference on Multimedia, pp 1337–1345.

Salvi, D., Bestagini, P., & Tubaro, S. (2023). Reliability Estimation for Synthetic Speech Detection. In *Proc. ICASSP*).

Santos, C., dos Santos, E. M., & Souto, E. (2012, July). Nudity detection based on image zoning. In 2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA) (pp. 1098-1103).

Sarridis, I., Koutlis, C., Papadopoulou, O., Papadopoulos, S. (2022, December). Leveraging large-scale multimedia datasets to refine content moderation models. In 2022 IEEE Eighth International Conference on Multimedia Big Data (BigMM) (pp. 125-132). IEEE

Saxena, A., Ajit, A., Arora, C., & Raj, G. (2023, March). Efficient Net V2 Algorithm-Based NSFW Content Detection. In International Conference on Information Technology (pp. 343-355). Singapore: Springer Nature Singapore.

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 815-823).

Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A., (2021). Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. arXiv.

Shaar, S., Babulkov, N., Da San Martino, G., & Nakov, P. (2020). That is a Known Lie: Detecting Previously Fact-Checked Claims. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online (pp. 3607–3618). Association for Computational Linguistics.

Shankar, S., et al. Generalizing across domains via cross gradient training. ICLR 2018.

Shih, K. J., Valle, R., Badlani, R., Lancucki, A., Ping, W., & Catanzaro, B. (2021, June). RAD-TTS: Parallel flow-based TTS with robust alignment learning and diverse synthesis. In ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models.

Shu, K., Cui, L., Wang, S., Lee, D., & Liu, H. (2019, July). defend: Explainable fake news detection. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 395-405).

Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., Kiela, D. (2022). Flava: A foundational language and vision alignment model. CVPR.

Song, Y., et al. ObjectStitch: Generative Object Compositing, arXiv:2212.00932, 2022.

Sultani, W., Chen, C., Shah, M., (2018). Real-world anomaly detection in surveillance videos. CVPR.

Tak, H., Patino, J., Todisco, M., Nautsch, A., Evans, N., & Larcher, A. (2021). End-to-end anti-spoofing with RawNet2. In ICASSP (pp. 6369-6373).

Tan, X., Chen, J., Liu, H., Cong, J., Zhang, C., Liu, Y., ... & Liu, T. Y. (2024). Naturalspeech: End-to-end text-to-speech synthesis with human-level quality. IEEE Transactions on Pattern Analysis and Machine Intelligence.

Tang, Z., et al. SelfNorm and CrossNorm for out-of-distribution robustness. ICCV 2021.

Teyssou, D., Leung, J.-M., Apostolidis, E., Apostolidis, K., Papadopoulos, S., Zampoglou, M., Papadopoulou, O., & Mezaris, V. (2017). The InVID Plug-in: Web Video Verification on the Browser. In Proceedings of the First International Workshop on Multimedia Verification (MuVer '17). Association for Computing Machinery, New York, NY, USA, 23–30.

Thomee, B., Shamma, D. A., Friedland, G., Elizalde, B., Ni, K., Poland, D., ... Li, L. J. (2016). Yfcc100m: The new data in multimedia research. *Communications of the ACM*, *59*(2), 64-73.

Thorne, J., Vlachos, A., Christodoulopoulos, C., & Mittal, A. (2018). FEVER: a large-scale dataset for fact extraction and VERification. arXiv preprint arXiv:1803.05355.

Tian, Y., Pang, G., Chen, Y., Singh, R., Verjans, J.W., Carneiro, G., (2021). Weakly-supervised video anomaly detection with robust temporal feature magnitude learning. ICCV.

Tolosana, R., Vera-Rodriguez, R., Fierrez, J., Morales, A., & Ortega-Garcia, J. (2020). Deepfakes and beyond: A survey of face manipulation and fake detection. Information Fusion, 64, 131-148.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., & Jégou, H. (2021, July). Training data-efficient image transformers & distillation through attention. In International conference on machine learning (pp. 10347-10357). PMLR.

Van der Goot, R., Üstün, A., Ramponi, A., Sharaf, I., & Plank, B. (2021). Massive Choice, Ample Tasks (MaChAmp): A Toolkit for Multi-task Learning in NLP. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Online (pp. 176–197). Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Viikki, O., & Laurila, K. (1998). Cepstral domain segmental feature vector normalization for noise robust speech recognition. Speech Communication, 25(1-3), 133-147.

Vovk, I., Sadekova, T., Gogoryan, V., Popov, V., Kudinov, M. A., & Wei, J. (2022). Fast Grad-TTS: Towards Efficient Diffusion-Based Speech Generation on CPU. In Interspeech (pp. 838-842).

Voynov, A., et al. Sketch-Guided Text-to-Image Diffusion Models, arXiv:2211.13752, 2022.

Wang, X., Yamagishi, J., Todisco, M., Delgado, H., Nautsch, A., Evans, N., ... & Ling, Z. H. (2020). ASVspoof 2019: A large-scale public database of synthesized, converted and replayed speech. *Computer Speech & Language*, 64, 101114.

Wang, X., & Yamagishi, J. (2022). Investigating self-supervised front ends for speech spoofing countermeasures. Odyssey 2022: The Speaker and Language Recognition Workshop.

Wang, G., Yuan, X., Zheng, A., Hsu, H.M., Hwang, J.N., (2019). Anomaly candidate identification and starting time estimation of vehicles from traffic videos. in CVPR workshops

Wang, Z., et al. Learning to diversify for single domain generalization. ICCV 2019.

Wu, P., Liu, J., Shi, Y., Sun, Y., Shao, F., Wu, Z., Yang, Z. (2020). Not only look, but also listen: Learning multimodal violence detection under weak supervision. ECCV.

Xie, Yuankun, et al. "An Efficient Temporary Deepfake Location Approach Based Embeddings for Partially Spoofed Audio Detection." arXiv preprint arXiv:2309.03036 (2023).

Xuan, H., Stylianou, A., & Pless, R. (2020). Improved embeddings with easy positive triplet mining. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 2474-2482).

Xue, Z., Liu, Q., Shi, H., Zou, R., & Jiang, X. (2022). A transformer-based DeepFake-detection method for facial organs. Electronics, 11(24), 4143.

Xie, Y., Cheng, H., Wang, Y., Ye, L. (2023) Learning A Self-Supervised Domain-Invariant Feature Representation for Generalized Audio Deepfake Detection. In *Proc. Interspeech*.

Xu, H., Ghosh, G., Huang, P.Y., Okhonko, D., Aghajanyan, A., Metze, F., Zettlemoyer, L., Feichtenhofer, C. (2021). Videoclip: Contrastive pretraining for zero-shot video-text understanding. EMNLP

Xu, X., et al. Prompt-free diffusion: Taking ”text” out of text-to-image diffusion models, arXiv:2305.16223, 2023.

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., & Raffel, C. (2021). mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online (pp. 483-498). Association for Computational Linguistics.

Xue, H., Guo, S., Zhu, P., & Bi, M. (2023). Multi-GradSpeech: Towards Diffusion-based Multi-Speaker Text-to-speech Using Consistent Diffusion Models. arXiv preprint arXiv:2308.10428.

Yang, B., et al. Paint by example: Exemplar-based image editing with diffusion models. CVPR 2023.

He, Yanzhang, et al. "Streaming end-to-end speech recognition for mobile devices." ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019.

Yao, Zhuoyuan, et al. "Wenet: Production oriented streaming and non-streaming end-to-end speech recognition toolkit." arXiv preprint arXiv:2102.01547 (2021).

Yaroshchuk, A., Papastergiopoulos, C., Cuccovillo, L., Aichroth, P., Votis, K., & Tzovaras, D. (2023, December). An Open Dataset of Synthetic Speech. In 2023 IEEE International Workshop on Information Forensics and Security (WIFS) (pp. 1-6). IEEE

Ye, Z., Xue, W., Tan, X., Chen, J., Liu, Q., & Guo, Y. (2023, October). Comospeech: One-step speech and singing voice synthesis via consistency model. In Proceedings of the 31st ACM International Conference on Multimedia (pp. 1831-1839)

Yi, J., Bai, Y., Tao, J., Ma, H., Tian, Z., Wang, C., ... & Fu, R. (2021). Half-truth: A partially fake audio detection dataset. In *Proc. Interspeech.*

Yu, F., et al. BDD100K: A diverse driving dataset for heterogeneous multitask learning. CVPR 2020.

Yue, X., et al. Domain randomization and pyramid consistency: simulation-to-real generalization without accessing target domain data. ICCV 2019.

Wadden, D., Lin, S., Lo, K., Wang, L. L., van Zuylen, M., Cohan, A., & Hajishirzi, H. (2020, November). Fact or Fiction: Verifying Scientific Claims. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 7534-7550).

Wang, M., Xing, J., Liu, Y. (2021). Actionclip: A new paradigm for video action recognition. arXiv.

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N. & Zhou, M. (2020). MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. Advances in Neural Information Processing Systems, 33, pp. 5776-5788.

Wang, W. Y. (2017, July). “Liar, Liar Pants on Fire”: A New Benchmark Dataset for Fake News Detection. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers) (pp. 422-426).

Wright, D., & Augenstein, I. (2020). Claim Check-Worthiness Detection as Positive Unlabelled Learning. In Findings of the Association for Computational Linguistics: EMNLP 2020, Online (pp. 476–488). Association for Computational Linguistics.

Wu, A., and Deng, C., Single-domain generalized object detection in urban scenes via cyclic-disentangled self-distillation. CVPR 2022.

Wu, Y., Tan, X., Li, B., He, L., Zhao, S., Song, R., Qin, T., & Liu, T. (2022). AdaSpeech 4: Adaptive Text to Speech in Zero-Shot Scenarios. , 2568-2572. https://doi.org/10.48550/arXiv.2204.00436.

Wu, P., Liu, J., (2021). Learning causal temporal relation and feature discrimination for anomaly detection. IEEE Transactions on Image Processing.

Wu, J.C., Hsieh, H.Y., Chen, D.J., Fuh, C.S., Liu, T.L. (2022). Self-supervised sparse representation for video anomaly detection. ECCV.

Zagoruyko, S., Komodakis, N. (2016). Wide residual networks. *arXiv preprint arXiv:1605.07146*.

Zampieri, M., Malmasi, S., Nako, P., Rosenthal, S., Farra, N., & Kumar, R. (2019). Predicting the Type and Target of Offensive Posts in Social Media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, *Volume 1 (Long and Short Papers)*, pages 1415–1420, Minneapolis, Minnesota. Association for Computational Linguistics.

Zampieri, M., Nakov, P., Rosenthal, S., Atanasova, P., Karadzhov, G., Mubarak, H., Derczynski, L., Pitenis, Z., & Çöltekin, C. (2020). SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1425–1447, Barcelona (online). International Committee for Computational Linguistics.

Zampoglou, M., Papadopoulos, S., Kompatsiaris, Y., Spangenberg, J. (2016, December). A web-based service for disturbing image detection. In *International Conference on Multimedia Modeling* (pp. 438-441). Cham: Springer International Publishing.

Zanella, L., Liberatori, B., Menapace, W., Poiesi, F., Wang, Y., & Ricci, E. (2023). Delving into CLIP latent space for Video Anomaly Recognition. ArXiv, abs/2310.02835.

Zaheer, M.Z., Mahmood, A., Khan, M.H., Segu, M., Yu, F., Lee, S.I., (2022). Generative cooperative learning for unsupervised video anomaly detection. CVPR.

Zeng, Y., et al. SceneComposer: Any-Level Semantic Image Synthesis, arXiv:2211.11742, 2022.

Zhang, L., Wang, X., Cooper, E., Evans, N., & Yamagishi, J. (2022c). The PartialSpoof database and countermeasures for the detection of short fake speech segments embedded in an utterance. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 31, 813-825.

Zhang, S., Zhu, X., Lei, Z., Shi, H., Wang, X., & Li, S. Z. (2017). S3fd: Single shot scale-invariant face detector. In Proceedings of the IEEE international conference on computer vision (pp. 192-201).

Zhang, Z., Zhao, X., & Yi, X. (2022a). ASLNet: An Encoder-Decoder Architecture for Audio Splicing Detection and Localization. Security and Communication Networks. https://doi.org/10.1155/2022/8241298.

Zhang, J., Zhao, Y., Saleh, M., & Liu, P. (2020, November). Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In the International conference on machine learning (pp. 11328-11339). PMLR.

Zhang, K., Liang, S., Nie, S., He, S., Pan, J., Zhang, X., Ma, H., & Yi, J. (2022b). A Robust Deep Audio Splicing Detection Method via Singularity Detection Feature. ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2919-2923. https://doi.org/10.1109/icassp43922.2022.9746596.

Zhang, X., Yi, J., Tao, J., Wang, C., & Zhang, C.Y. (2023). Do You Remember? Overcoming Catastrophic Forgetting for Fake Audio Detection. In *Proc. ICML*.

Zhao, H. Y. Chen, R. Wang, and H. Malik (2017), “Audio Splicing De- tection and Localization Using Environmental Signature,” Multi- media Tools and Applications, vol. 76, no. 12, pp. 13 897–13 927, 2017.

Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M. (2023). Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Zhao, Y., et al. Style-Hallucinated Dual Consistency Learning: A Unified Framework for Visual Domain Generalization, International Journal of Computer Vision, 132(3):837–853, March 2024.

Zi, B., Chang, M., Chen, J., Ma, X., & Jiang, Y. G. (2020, October). Wilddeepfake: A challenging real-world dataset for deepfake detection. In Proceedings of the 28th ACM international conference on multimedia (pp. 2382-2390).

Zhong, Z., et al. Adversarial style augmentation for domain generalized urban-scene segmentation. NeurIPS 2022.

Zhou, K., Yang, J., Loy, C.C., Liu, Z. (2022). Learning to prompt for visionlanguage models. IJCV.

Zhou, K., et al. Domain generalization with mixstyle. ICLR 2021

Zhu, J., et al. Unpaired image-to-image translation using cycle-consistent adversarial networks. ICCV 2017.