FERMI

FAKE NEWS
RISK MITIGATOR

Project acronym:	FERMI
Project full title:	Fake nEws Risk Mitigator
Call identifier	HORIZON-CL3-2021-FCT-01
Start date:	01/10/2022
End date:	30/09/2025
Grant agreement no:	101073980

D3.2 FERMI Technology Facilitators Package

Work package: 3	
Version: 1	
Deliverable type: Report	**Dissemination level:** Public
Official submission date: M30	**Actual submission date:** M36

Funded by
the European Union

Leading author(s):

Surname	First name	Beneficiary
Dugato	Marco	UCSC-Transcrime
Lo Giudice	Michael Victor	UCSC-Transcrime
Tsakova	Gergana	UCSC-Transcrime

Contributing partner(s):

Surname	First name	Beneficiary
Papadakis	Athanasios	Netcompany - Intrasoft
Gousetis	Nikolaos	Netcompany - Intrasoft
Troulitaki	Petrina	ITML
Stamatis	Giorgos	ITML
Dimakopoulos	Nikos	ITML
García Gómez	Joaquín	ATOS
Kolliarakis	Georgios	BIGS
Baronchelli	Adelaide	BIGS
Valente	Catarina	INOV
Varela da Costa	Joao	INOV

Peer reviewer(s):

Surname	First name	Beneficiary
Vrotsou	Christina	CONV
Sven-Eric	Fikenscher	BPA

Ethics reviewer:

Surname	First name	Beneficiary
Giglio	Flavia	VUB

Security reviewer:

Surname	First name	Beneficiary
Bourgos	Paraskevas	INTRA

Document Revision History

Version	Date	Modifications Introduced	
		Modification Reason	Modified by
0.05	02.08.2024	Table of contents and draft structure formulated and shared with partners	UCSC-Transcrime

0.15	**31.10.2024**	**Provision of WP3 partners' first contributions**	**UCSC-Transcrime; INTRA; ITML; ATOS**
0.25	**15.12.2024**	**Provision of WP3 partner's second contributions**	**UCSC-Transcrime; INTRA; ITML; ATOS**
0.35	**14.02.2025**	**Provision of WP3 partner's third contributions**	**UCSC-Transcrime; INTRA; ITML; ATOS**
0.40	**29.05.2025**	**Provision of first draft of integrated version deliverable**	**UCSC-Transcrime**
0.50	**10.06.2025**	**Edits and feedback from WP3 partner's on first draft of integrated version deliverable**	**INTRA; ITML; ATOS; BPA**
0.60	**23.06.2025**	**Provision of second draft of integrated version deliverable**	**UCSC-Transcrime**
0.65	**16.07.2025**	**Initial ethics review**	**KUL**
0.65	**16.07.2025**	**Initial peer review**	**CONV**
0.75	**28.07.2025**	**Provision of third draft of integrated version deliverable**	**UCSC-Transcrime**
0.80	**28.07.2025**	**Final ethics review**	**KUL**
0.80	**01.09.2025**	**Final peer review**	**BPA**
0.85	**19.09.2025**	**Initial security review**	**INTRA**
0.90	**20.09.2025**	**Provision of fourth draft of integrated version deliverable**	**UCSC-Transcrime**
1	**29.09.2025**	**Completion of final document**	**UCSC-Transcrime; BPA**

Executive Summary

The D3.2 Technology Facilitators' Package – 2nd Version aims to provide an in-depth overview of the technologies being developed by the Fake News Risk Mitigator consortium in their *end-product* or *final* state, their adherence to commitments made in the Grant Agreement (Work Package 3, Tasks 3.1, 3.2, 3.4, and 3.6), and the adjustments made based on end-user feedback following the piloting of the technologies. Moreover, this deliverable seeks to outline the versatility of said technologies to changing end-user needs following the lifetime of the FERMI project.

In their final state of development, these technologies provide end-users with a wholistic understanding of a given disinformation or fake news event's content, spread, and impact. Through innovative applications of machine learning and artificial intelligence, as well as swarm learning, ensuring data-privacy is protected, end-users gain a greater intelligence picture, including an understanding of if the disinformation campaign they are investigating is being propagated by human beings or a bots, the online network of interactions, the content and conversation's sentiment, and how offline-crime occurrences are probably to evolve in the 4-week period following the start of the investigation.

T3.1, **the Dynamic Flows Modeler**, is an AI-driven crime prediction device which, built using big-data, natural language processing and machine learning, generates informed estimates for the impact of an online disinformation or fake news event on the number of offline crime occurrences in NUTS2 regions of Europe. Specifically, the device employs one-dimensional convolutional neural network architecture granting the capability to wholistically forecast the likely evolution of offline crimes in a given area, for a 4-week period once provided a contemporary disinformation or fake news event.

T3.2, **the Spread Analyser,** consists of three main functionalities that capture the spread of a given disinformation event, on social media, among other accounts, which of these are most influential, and whether said accounts are controlled by humans or operated by bots. The component, starting from the user-provided post, builds a graph depicting the disinformation spread related to the investigated post. This process maps how the investigated post was propagated amongst other users and showcases the network of disinformation throughout the platform. Furthermore, for each given post in the graph, the component provides complimentary details on said post and the poster, including the poster's public metrics. Subsequently, the application of machine learning models and graph analysis services produce insights regarding the given post's influence on the network, identifies the probable origin post, and the user's classification as human controlled or bot operated.

T3.4 involved the construction of a federated learning paradigm, characterised by implementing decentralised training of machine learning algorithms. Specifically, T3.4 employed a variant of said methodology, swarm learning. **Swarm learning** allows different providers of data to obtain a common model without needing to share private data with each other, maintaining privacy. In the context of FERMI, this involves the pooling of data from several law enforcement agencies without violating data protection norms and rules while empowering the Dynamic Flows Modeler by providing the needed context of past criminal behaviour in forecasting future crime.

T3.6, **the Sentiment Analysis module,** emerges as a valuable component within the FERMI project. Designed to examine disinformation and fake news within social media posts, the Sentiment Analysis Module provides end-users with insights into the emotional tone of the content. This module employs advanced natural language processing techniques, leveraging bidirectional encoder representations from transformers, as outlined in the Grant Agreement. By incorporating bidirectional context—where the classification of an instance is informed by both past and future instances—the module offers a comprehensive understanding of sentiment trends in the analysed content. This functionality supports end-users in effectively interpreting the emotional undertones present in social media discussions. To ensure privacy and data protection as well as relevance, the module anonymises the posts, removes links, and processes emoji characters to export additional information.

D3.2 details the **integration between the components above and T3.5 and T3.3,** the **Behaviour Profiler & Socioeconomic Analyser** and the **Community Resilience Management Modeler**. T3.5 aims to quantify likelihood and severity of crimes occurring due to disinformation whose combined terms outputs a measurement of impact. The former, T3.3, the Community Resilience Management Modeler, seeks to support law enforcement agencies in their decisions in regards to countering disinformation online and the potential adverse effects it has on crime and society as a whole. It does so by offering countermeasures, specifically with respect to resource allocation. The integration between these technologies and the tasks centric to D3.2 is, specifically, through the Dynamic Flows Modeler, which provides its output to the Behaviour Profiler & Socioeconomic Analyser. The Dynamic Flows Modeler's output is, thus, the input with which T3.3 and T3.5 begin operations.

Table of Contents

Abbreviations

1-D CNN:	One Dimensional Convolutional Neural Networks
API:	Application Programming Interface
ARIMA:	Autoregressive Integrated Moving Average
BERT:	Bidirectional Encoder Representations from Transformers
BFP:	Belgian Federated Police
BPA:	Bavarian University of Public Service
CNN:	Convolutional Neural Network
D&FN:	Disinformation and Fake News
FERMI:	Fake News Risk Mitigator
FL:	Federated Learning
FMI:	Finland Ministry of the Interior
GA:	Grant Agreement
GDP:	Gross Domestic Product
GDS:	Graph Data Science
GOP:	Grand Old Party
GRU:	Gated Recurrent Unit
KPI:	Key Performance Indicator
KR:	Key Result
LEA:	Law Enforcement Agency / Agencies
LSTM:	Long Short-Term Memory
MAE:	Mean Absolute Error
mBERT:	Multilingual Bidirectional Encoder Representations from Transformers
ML:	Machine Learning
MLP:	Multilayer Perception
NLP:	Natural Language Processing
NUTS:	Nomenclature of Territorial Units for Statistics
PU:	Public
ReLU:	Rectified Linear Unit
RMSE:	Root Mean Square Error
RNN:	Recurrent Neural Network
SGD:	Stochastic Gradient Descent
SL:	Swarm Learning
SOTA:	State-of-the-Art
SST:	Stanford Sentiment Treebank
SST3:	3 Class Stanford Sentiment Treebank
TOT:	Technically Oriented Targets
TRL:	Technological Readiness Level
XLM-R:	Cross-lingual Langugae Model – RoBERTa

Technologies' Abbreviations:

Task	Grant Agreement Name	Abbreviation in D3.2
T3.1	D&FN-induced and D&FN-enabled offline crimes analysis	Dynamic Flows Modeler
T3.2	Disinformation Sources and Spread Analysis and Impact Assessment	Spread Analyser
T3.4	Swarm Learning framework	SL
T3.6	The sentiment analysis module	Sentiment Analysis module

1 Introduction

Deliverable 3.2, the technology facilitators package second version, provides an in-depth review of most of the technological components in the Fake News Risk Mitigator (FERMI) project, specifically tasks T3.1, T3.2, T3.4 and T3.6. The deliverable focuses on highlighting said technologies' compliance with the Grant Agreement (GA), wherein commitments were made regarding the technologies' development, function, and performance. As a sequel deliverable to Deliverable 3.1, Deliverable 3.2 speaks to the end-product version of each component, that is, the components as they are at the end of the FERMI project. **The technologies have been improved following an initial round of pilot testing, with feedback from pilot-users, and the input of the reviewers during the project's mid-term review**. Just as well, the FERMI consortium itself identified next steps for the technologies, as detailed in Deliverable 3.1, and a set of technically oriented targets (TOTs), self-imposed, to ensure the outcomes of the FERMI project were the highest calibre possible given the time-frame of the project.

Four technological components, independently developed but now successfully integrated, are featured: (T3.1) D&FN-induced and D&FN-enabled offline crimes analysis, henceforth referred to as the Dynamic Flows Modeler; (T3.2) Disinformation Sources and Spread Analysis and Impact Assessment, henceforth the Spread Analyser; (T3.4) the Swarm Learning (SL) infrastructure/framework, for holistic AI-based services in law enforcement agencies (LEA), and (T3.6) the Sentiment Analysis module.

The **Dynamic Flows Modeler (Section 2)** serves to "evaluate the degree in which the spread of [disinformation and fake news (D&FN)] online impacts on the occurrence of offline crime,"[1] an analysis which "will evaluate the intensity of the relation between the spread of D&FN and offline crimes, the temporal patterns in the relation, [and] the spatial decay of the relation."[2] Moreover, the Dynamic Flows Modeler "produce[s] AI-based [estimates] of the most likely spatiotemporal evolution of D&FN-induced and D&FN enabled offline crimes."[3] **The Spread Analyser (Section 3)**, instead, is "a tool that will take as input news already classified as [D&FN] and will be able to trace and map this news to their main actors/accounts which are responsible for creating and spreading the [D&FN] across the network."[4] The Spread Analyser, in accordance with the GA, can classify if the identified actors/accounts are physical persons or bots and assign an influence index to their role/power over the network. T3.4, **the SL framework (Section 4)**, involved the creation of "a scalable software architecture for training Machine Learning models near to the data sources where they are generated."[5] In other words, through the development of this swarm learning technology, the FERMI platform, particularly tools such as the Dynamic Flows Modeler, is equipped with past crime data from multiple, independent LEA partners while not violating their obligation to preserve and protect the personal data of those involved in past criminal events, having said data never leave their secure severs. **The Sentiment Analysis Module (Section 5)**, analyses D&FN, specifically in social media posts, to provide end-users a perception of the emotional tone in said posts' content. The Sentiment Analysis Tool, in accordance with the GA, exploits bidirectional encoder representations from Transformers (BERT). In doing so, by "the classification [of] results of one specific instance are affected by both past and future instances,"[6] providing end-users a wholistic understanding of the content's sentiment. The analysis of social media post is preceded by ensuring the anonymisation of the posts, deletion of links, and replacing of emoji characters with corresponding text/keyword.

An important factor in understanding the aforementioned technologies' function, with the greater FERMI platform, is how they are planned to be integrated with the two other tasks of Work Package 3, T3.3 and T3.5, that is, the Community Resilience Management Module and the Behaviour Profiler/Socioeconomic Analyser. These further technologies are downstream, in terms of the flow of data, from the integrated technologies featured here; therefore, how the various outputs of these technologies are passed to T3.3 and

[1] 'Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021.
[2] Ibid.
[3] Ibid.
[4] Ibid.
[5] Ibid.
[6] Ibid.

T3.5, and how they share the broader infrastructure is addressed. These points, in addition to how the integration complies with the GA is well articulated in the sixth section.

Sections 2 through 5 are structured as follows, for each section, and therefore technological component, a practical description of the components function is provided, with emphasis placed on the improvements made between the first and end-product versions. With a specific subsection highlighting the feedback received from pilot-users and how it was addressed. Then, a technical description adds nuance to the practical description, providing insights on the specific technical operation of the component, again with emphasis placed on changes made after the submission of Deliverable 3.1. **Two additional subsections conclude each component's segment of the deliverable, recounting, importantly so, the achievement of the performance metrics assigned to the component and the versatility of the component to changing end-user needs**, a reality that characterises the constantly evolving reality of law enforcement and hybrid threats such as D&FN.

As has been indicated in the aforementioned paragraphs, **Deliverable 3.2 should be read as the sequel to Deliverable 3.1 and, as such, expands in great detail on the changes and improvements made between deliverables,** avoiding a too-in-depth retelling of what has remained static. That being said, Deliverable 3.2, nonetheless, provides a wholistic overview of each technological components. Components which **offer to LEAs a new degree of analytical insights through a direct linkage with social media platforms**, as opposed to merely evaluating a social media post unto itself. The advantage of this linkage is that the FERMI platform can provide a grasp as to the movement of a disinformation post through social media, understanding the influence of the accounts interacting with it. As stated in Deliverable 3.1, a post-centric analysis, where no linkage with platforms is established, interactions have a two-dimensional appearance, where every interaction carries the same weight. Evidently, this is not the case, as certain users have far greater pull than others. Since the submission of Deliverable 3.1, **all components developed within the FERMI project have shifted to being social media agnostic**, meaning that despite certain social media networks were relied on for testing and development, the tools have been adapted such that **they may be adjusted and applied to any social media platform, should access become available**. Indeed, the end-product versions of **the technologies have been successfully employed with both X and Mastodon's API.**

2 Task 3.1 – The Dynamic Flows Modeler

T3.1, **the Dynamic Flows Modeler, is an AI-driven crime estimation device which, utilising big-data, natural language processing and machine learning, was trained to create informed estimates of offline crime occurrences**, allowing for the impact of an online disinformation or fake news event in NUTS2 regions of Europe to be better understood and, in turn, provide insight to law enforcement agencies when deciding how to allocate their resources. **Through repeated attempts at improvement, the device, which previously combined various machine and deep learning approaches, was simplified, now relying exclusively on one-dimensional convolutional neural networks** (1D-CNN) in producing estimates for the number of crime occurrences in a given area, for the four weeks following the date an end-user begins an investigation. The Dynamic Flows Modeler's **estimates are wholistic**, meaning they provide a likely level of crime considering a range of factors, primarily the historical levels of crime (provided via FERMI's SL framework) and the online spread of the disinformation post being investigated (delivered by the Spread Analyser).

After thorough testing, the Dynamic Flows Modeler can be applied to two topics of disinformation, covering all three of FERMI's use-cases: COVID-19/public health and political extremism (comprised of left- and right-wing extremism). Importantly, between the submission of Deliverable 3.1 and the writing to this deliverable, there has been significant progress in terms of the architecture which empowers the Dynamic Flows Modeler. Reduction in the number of deep learning architectures being relied on, improvement in estimation precision, changes to the user-interface, and the reduction of time needed to produce results resulting in further compliance with GA specified targets.

Subsection 2.1 will begin with an overview of the Dynamic Flows Modeler's basic function, then proceeding to cover the pilot feedback, specific inputs and outputs to the module, and the changes made between Deliverable 3.1 and Deliverable 3.2. As such, 2.1 provides an understanding of the end-product Dynamic Flows Modeler. Subsection 2.2, instead, illustrates the model from a technical point of view – explaining the backend operations and architecture, as well as reviewing its development stages. 2.3 presents the key performance indicators (KPI), key results (KR) and TOTs assigned to the Dynamic Flows Modeler and its cohesion with them. Before concluding, Subsection 2.4 seeks to make clear how the Dynamic Flows Modeler can be adapted and persist in the face of changing end-user needs in the years to follow the FERMI project.

Work Package 3's Task 3.1, covered in this section, featured a commitment to acquire micro-level data for real crime occurrences, offline, and on D&FN, for the purpose of evaluating the relationship between D&FN and offline crime. Moreover, the GA states a necessity for the Dynamic Flows Modeler to make AI-based estimates regarding the spatiotemporal occurrences of D&FN-induced and -enabled offline crime in future periods. How these GA commitments are, or will be, fulfilled by the Dynamic Flows Modeler will be touched on throughout all the proceeding sections.

2.1 Practical Description

The Dynamic Flows Modeler**,** has advanced beyond a device capable of solely evaluating the relationship between offline crime and the spread of online D&FN such as to make forecasts of future crime occurrences, to **now being accessible to an end-user and presenting its results within a broader context**. Following several rounds of refinement, the Dynamic Flows Modeler, produces an estimate of the evolution of four crime types over the next four-week period at a NUTS2 level, following an online D&FN event. Each estimate is accompanied by a rate of change percentage, allowing the end-user to easily understand the direction and relative magnitude from the level of crime at the present (specifically, week 1) to the estimated future period.

The four crime types were selected on the basis of accuracy (how well the model performed), availability of data (due to jurisdictional differences in how crimes were defined), and relevancy to D&FN. As will be explained in the proceeding sections, financial crimes and crimes unlikely to be driven by extremist online content were omitted, with the inclusion of those crimes that either imply the immediate use of violence or at least appear to imply a certain proneness to violence, in line with the FERMI project's intention to examine the ramifications of D&FN-informed violent extremism.

As in accordance with the GA, the Dynamic Flows Modeler meets the objective of merging and organising data in order "to evaluate the degree in which the spread of D&FN online impacts on the occurrence of offline crime" capturing the intensity of the relationship and its temporal patterns (i.e., spatial decay) within its wholistic prediction of crime, informed of/with the spread of D&FN via the Spread Analyser. Just as well, this is accomplished through AI-based methods, specifically 1-D CNN deep learning algorithms.

In the proceeding subsections, the following aspects of the Dynamic Flows Modeler will be covered at greater depth: the input and output of the Dynamic Flows Modeler (2.1.1); the feedback received in the second FERMI pilot and how it has been addressed in the current end product (2.1.2), and, lastly, the situation of the Modeler within the FERMI platform (2.1.3).

2.1.1 The Dynamic Flows Modeler's Input and Output

2.1.1.1 Input

The Dynamic Flows Modeler is best described as being comprised of 8 different deep learning models, two for each crime type, one for the political extremism use-cases, and one for the public health threat use-case. Thus, there is a model specifically trained to predict each unique crime, having studied said crimes occurrences and the level of the spread of disinformation related to the given use cases. Therefore, the models are the following: (1) assault – political extremism; (2) assault - COVID-19/public health; (3) destruction/damage/vandalism of property – political extremism, (4) destruction/damage/vandalism of property – COVID-19/public health; (5) disorderly conduct – political extremism; (6) disorderly conduct - COVID-19/public health; (7) larceny/theft – political extremism, and (8) larceny/theft – COVID-19/public health.

Each of these models requires the same input in order to produce estimates for future crime occurrences, specifically, past crime occurrences for the 14 prior weeks, including the week in which the estimate is being requested, and the spread of the disinformation post being investigated, recorded at a weekly level. It is noteworthy that while the initial version of the Dynamic Flows Modeler required a wide variety of socio-economic inputs to produce estimates, the second version has moved away from this dependence, successfully relying on past crime levels to inform the model as to varying socio-economic indicators. Crime is, indeed, a product of socio-economic values, thus changes to the socio-economic reality of a NUTS2 region, relevant to estimating future crime occurrences, are captured within the past crime data. This choice is validated in the reduced mean absolute error (MAE) of the Dynamic Flows Modeler upon the shift away from the over provision of socio-economic values.

Past crimes are provided to the Dynamic Flows Modeler via the SL framework, while the spread of the D&FN post being investigated is compiled and shared by the Spread Analyser. Both components and their operation are explained in further detail in the proceeding sections, here, however, it is important to note that the SL framework allows for past crime data to be shared with the Dynamic Flows Modeler via a federated learning trained machine learning model. As such, the SL framework ensures that the data never leaves the premises of the end-user LEA who rightfully possess said data and any gaps in the data provided, due to missing values or unavailability of data for the weeks immediately pre-dating the week of the estimation request, are rectified through the provision of an estimated value produced by a model trained on the private LEA data.

2.1.1.2 Output

These inputs provide the Dynamic Flows Modeler with the needed context to evaluate and produce an estimate for the proceeding four weeks. The estimates take the form of float values, reporting the number of crimes likely to occur for each crime type in each NUTS2, for the proceeding 4 weeks. Alongside these forecasts is a rate of change, in percentage, with the sign of said change (positive or negative) indicated with an arrow icon (see Figure 1). This rate of change is calculated by identifying the percent variation between the level of the same crime, in the same NUTS2, between week 1 (the week in which the prediction is made) and the week of each estimate. The formula is presented in Equation 1, where Δ is the rate of change, in week *t* (*Wt*) for crime type *i* (*Ci*):

$$\Delta\%_{W_t,C_i} = 100\left(\frac{W_t - W_1}{W_1}\right)$$

Equation 1: Dynamic Flows Modeler rate of change

The results are accessible through a map interface (explained in greater detail in subsection 2.3), with each NUTS2 region being coded with a colour (green, yellow, red) to represent the intensity of the change in crime's occurrence in the forecasted weeks (when the investigation is launched). Said intensity is calculated through a two-stage function: (1) deriving per crime type (*Ci*) an overall rate of change for the four weeks (W*t)* forecasted and (2) finding the mean of the overall rates of change of the four crimes, in a given region (*Ri*).

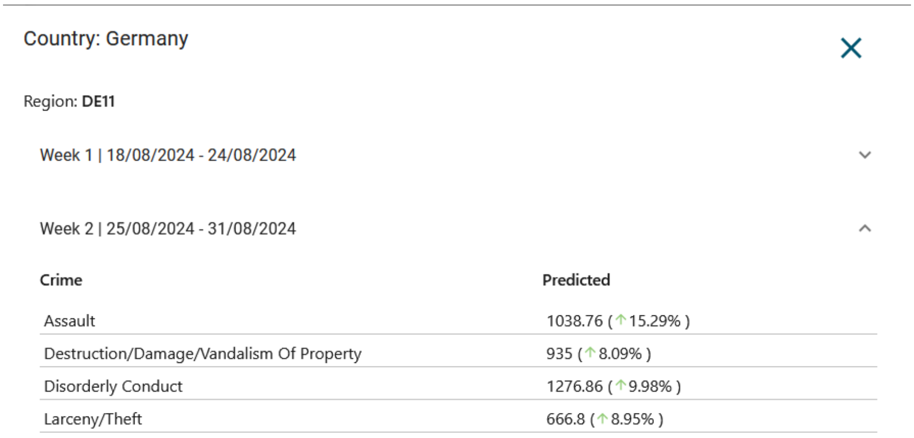
$$\Delta\%_{C_i,R_i} = \frac{1}{4}\sum_{W_t\,=\,2}^{5} \Delta\%_{W_t,C_i}$$

Equation 2: Overall crime type's rate of change in a NUTS2, over forecasted period

$$Intensity_{R_i} = \frac{1}{4}\sum_{C_i\,=\,1}^{4} \Delta\%_{C_i,R_i}$$

Equation 3: NUTS2 intensity of the rate of change over forecasted crime types and weeks

Where $\Delta\%_{C_i,R_i}$ represents the mean of the rate of change assigned to a specific crime type across all weeks, in a given region and $Intensity_{R_i}$ captures the average rate of change across crime types in a specific region. The colour code is then assigned based on the following rule: green when $Intensity_{R_i} \leq 25\%$; yellow when $25\% < Intensity_{R_i} \leq 75\%$, and red when $75\% < Intensity_{R_i}$.



Country: Germany

Region: **DE11**

Week 1 | 18/08/2024 - 24/08/2024

Week 2 | 25/08/2024 - 31/08/2024

Crime	Predicted
Assault	1038.76 (↑15.29%)
Destruction/Damage/Vandalism Of Property	935 (↑8.09%)
Disorderly Conduct	1276.86 (↑9.98%)
Larceny/Theft	666.8 (↑8.95%)

Figure 1 – Dynamic Flows Modeler output example

2.1.1.3 Output in Context

The Dynamic Flows Modeler has a wholistic approach, to ensure that end-users can understand the results in context, as within the training of the models (explained in greater detail in the proceeding section) weather patterns and seasonal realities (e.g., lower population density in the month of August) were included, moreover variation in pre-existing levels of crime have a significant impact, alongside the impact of disinformation, on the crimes to occur in the next 4 weeks. Therefore, provided estimates are for crime, taking into account the spread of the D&FN post provided by the end-user, but not solely the occurrences driven by disinformation. As such, the forecasted spatio-temporal evolution of crime is not, exclusively, driven by the provided D&FN post, instead, **end-users are always presented the *complete picture*** to ensure they are best informed on how to allocate resources in response to a D&FN campaign online, informed by the Dynamic Flows Modeler as well as the human intuition and experience that they already possess.

Moreover, the Dynamic Flows Modeler, through wholistic estimations, may forecast a decrease in crime, even if a D&FN campaign that is significantly likely to induce- and enable-offline crime is identified from the end-user provided social media post, seasonal factors or an existing downward trend in criminal activity may cause the estimated crime type to decrease, nonetheless. If, alternatively, the Dynamic Flows Modeler provided end-users with the number of crimes estimated to be induced by D&FN, exclusively, a problematic result may often arise, wherein the estimate shows an increase, as crime is higher than it would be without D&FN, yet, seasonal factors, or overall downward trends in a certain crime, may decrease the overall level of crime greater than the D&FN-induced increase. In such a situation, the Dynamic Flows Modeler's results may inspire a reallocation of resources towards a decreasing crime type, misrepresenting the overall evolution of criminal behaviour and potentially misguiding end-user LEAs. As such, the wholistic estimate was chosen and best informs end-users.

Just as well, **it is important to note that the output provided will be exclusively for the country and use-case requested when the investigation is launched on the FERMI platform**. In the first version of the Dynamic Flows Modeler, when a request for estimating future crime was sent to the model, a prediction was made for all crime types and in all NUTS2 regions of the pilot countries. This set up was computationally expensive, resulting in high latency and significant resources consumption to generate results. With the current arrangement, wherein the request is narrowed to a specific country and use-case, the Dynamic Flows Modeler requires a run-time to produce the estimates that is significantly shorter.

2.1.2 Pilot Feedback and D3.1's Outstanding Steps

Moving away from an overview and towards specifics, subsection 2.1.2 underlines where the first version of the Dynamic Flows Modeler needed improvement, in the opinion of pilot participants, and, more importantly, how these expectations were taken into account when developing the second version (i.e., the end product).

Table 1: Outstanding KPIs regarding pilot feedback

End-User Requirement	Pilot Evaluation KPI	Pilot 2, Round 2 Evaluation
UR014: The user is able to predict who are the potential victims of crimes related to D&FN.	>80% required	**95.65%**
UR017: The user can identify the geographical unit in which the criminal event may more likely occur due to the D&FN.	>65% required	**82.61%**
UR021: The user is able to identify potential threats to public safety.	>65% required	**95.65%**
UR026: The user is able to easily handle an AI-based tool to reliably predict the scope of disinformation-induced crimes.	>65% required	**95.65%**
UR031: The user should be able to access accurate information regarding offline crimes stemming from D&FN campaigns, improved through incoming data collected from different LEAs/sources.	>65% required	**91.30%**
UR038: The user is able to provide near real-time alerts and notifications to law enforcement	>65% required	**91.30%**

officers when new threats are detected. The alerts should be customised based on the user's preferences and job responsibilities.		

The feedback acquired from the pilot participants left six KPIs unmet by the first version of the Dynamic Flows Modeler, said KPIs are reported in the table above. In order to move the Dynamic Flows Modeler into alignment with said KPIs, and the desires of end-users, several changes were made to the device. Firstly, UR017, UR021, UR026, and UR038 revolve around the capacity of the Dynamic Flows Modeler to inform priority areas/persons for LEAs in the wake of a D&FN campaign and easily handle an AI-based tool, aiding in better resource distribution.

Changes to the user-interface and the backend were undertaken, where the first version focussed on presenting the accuracy of crime forecasts, the end product Dynamic Flows Modeler adds a layer of analysis, plotting its results onto a map interface at a NUTS2 level. As exhibited in Figure 2, **the results of the Dynamic Flows Modeler are presented in such a manner as to now quickly alert the end-user to the where and intensity in changes in offline crime**. As such, even before reviewing the individual crime forecast figures, the end-user gets a real-time understanding of the potential evolution of events yet to occur. These changes likewise contribute to addressing UR014, wherein the user is able to roughly predict who are the potential victims of crimes related to D&FN (an outstanding item that was planned on being addressed anyway, as explained in D3.1); due to the new map view, the end-user can understand which regions inhabitants are most at risk.

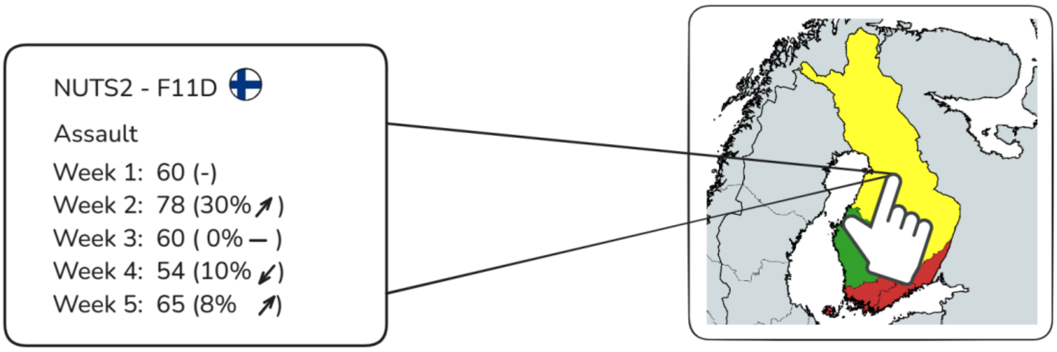


Figure 2: Mock example of the map interface for the Dynamic Flows Modeler

UR031, alternatively, deals with the end-user’s ability to understand the scope of D&FN-induced offline crime, improved through the incorporation of LEA data. As explained in the proceeding sections (see 2.4 on the connection with the SL framework and 4.2 on the achievement of accuracy-related KPIs), the end product version of the Dynamic Flows Modeler improves on its first version’s accuracy and features an API linkage with the SL Framework of T3.4 - enabling that LEA data (without ever having been directly shared) is enriching the produced forecasts.

2.1.2.1 Results of the Second Round of Pilots

Following the aforementioned changes, the pilot 2 session, of the second round of pilots, was conducted. With 23 total pilot-users (consisting of 7 active-duty LEA personnel, 5 non-active duty LEA staff, 11 LEA advisers, and 2 acquisition experts) and 23 completed questionnaires, the Dynamic Flows Modeler scored highly and meeting all UR satisfaction targets, as reported in Table 1. The marked improvements in pilot-user evaluations of predictive capability (UR014, UR017, UR026, and UR027), resource allocation (UR004), and accurate understand of offline crime (UR031) represent growing pilot-user confidence in the technological offerings and the success of the implemented changes in addressing pilot-user feedback from the first round of pilots. For further and more detailed information on the results of the second round of pilots, please refer to Deliverable 5.3 – The FERMI Final Execution Reports & Assessments.

2.1.3 The Dynamic Flows Modeler, Within FERMI

Within the FERM platform, the Dynamic Flows Modeler is situated with the Spread Analyser (T3.2) upstream and the Community Resilience Management Module (T3.3) downstream. Parallel to it is the SL framework (T3.4) and Sentiment Analysis Module (T3.6), whilst the Socio-economic Analyser and Behaviour Profiler (T3.5) is integrated within the Modeler.

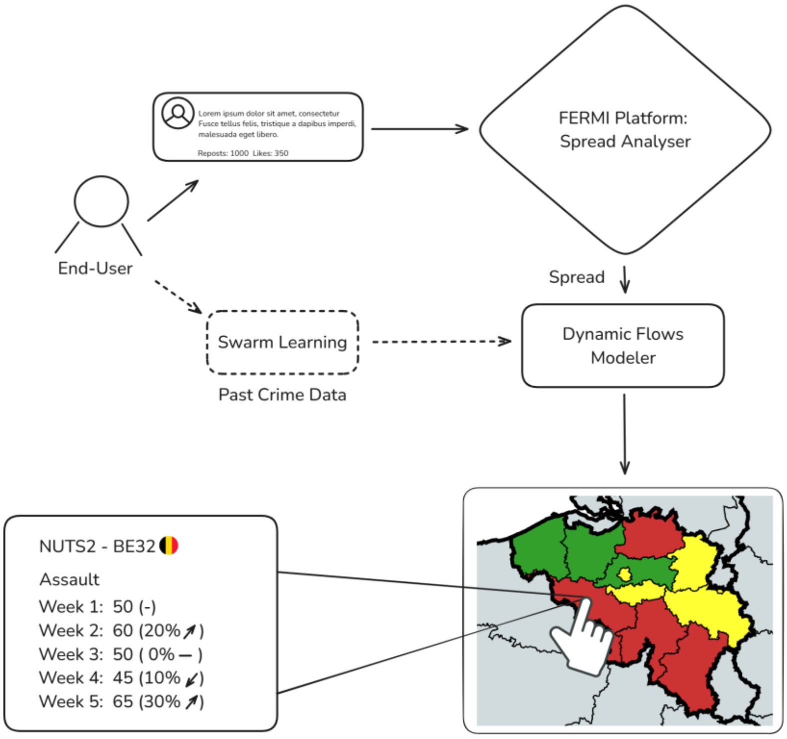


Figure 3: The Dynamic Flows Modeler within FERMI representation

As illustrated in Figure 3, end-user triggers the investigation by providing the platform with a social media post identified as disinformation. The Spread Analyser then passes to the Dynamic Flows Modeler a proxy measure for the spread of the post, online. Using X as an example, the Spread Analyser provides the Dynamic Flows Modeler with the number of reposts that occurred in the whole identified network for each week since the initial post was made. Then, the Dynamic Flows Modeler requests, via the SL framework (thus without compromising the privacy of data held on LEA end-users' servers) the level of past crime in the week preceding the investigation. In doing so the Dynamic Flows Modeler achieves the GA specification of "acquiring micro-level data on… actual offline criminal events from participant police authorities,"[7] while at the same time not having the privacy of said data be compromised. Using these two time-series, the estimated crime occurrences in the weeks going forward is produced. For the end-user, the results are made accessible via a map interface, with a colour coding (red, yellow, green) that allows for easy identification of where there is a greater risk/probability of offline crime, in line with feedback recounted in subsection 2.1.2. The colour coding reflects the intensity of the rate of change at an overall level within the region (across crime types and weeks). Proceedingly, explained in greater depth in Section 6, the estimated crime occurrences are inputted into the Socio-economic Analyser and Behaviour Profiler's formula which calculate impact of D&FN-induced crime in terms of economic cost. The results of said impact calculation are binned between 1 – 5, ordinal categories. The Dynamic Flows Modeler then communicates the impact score to the Community Resilience Management Modeler.

[7] Ibid.

2.2 Technical Description

The second version of the Dynamic Flows Modeler's technical operation can be described as having been simplified, while improving its accuracy. The first version (see Deliverable 3.1) featured three deep learning architectures, with all three making estimates for the evolution of offline crime and the device selecting the best performing, in terms of accuracy, to use for a given crime type. That is, for crime type X, the architecture that had the most accuracy in estimating said crime, in training and development, was to be incorporated into the FERMI platform as the architecture for predicting said crime. This meant the Dynamic Flows Modeler was characterised as several deep learning models placed together within a user-accessible device. In developing the second version, re-examination of the hyper-parameters employed, that is how the models were set to make estimates, lended to the decision to select a single deep learning architecture. A decision that also led to an improved accuracy and coherence with the GA and the TOT's objectives. In terms of data, both in how it was collected and in how it was pre-processed, subsection 3.1 will provide a brief overview as there are no significant deviations from the first version. For greater depth on the data employed, please refer to Deliverable 3.1.

2.2.1 Data: Collection and Pre-Processing

2.2.1.1 Past Crime Incidents

Past crime data, used to train the model, was collected from 30 American municipalities and 1 county. The data selection was based on availability, correspondence with available D&FN data, and the likelihood of a nexus between D&FN and the crime types. **The included data was published at an incident level (i.e., per crime occurrence), with the date of occurrence, and the type of crime specified**. Moreover, all the crime data had to be published publicly (i.e., open source) and be available for the same years as the collected D&FN data. **Open source publication of the data included ensures transparency** in the development process. The crime types were as categorised under the Federal Bureau of Investigation's universal crime reporting system. Financial/white collar crimes (i.e., various forms of fraud, bribery, counterfeiting, embezzlement, and commerce violations) and 'victimless' crimes (i.e., gambling, fugitive, immigration, pornography, treason, loitering, drunkenness, perjury, and non-violent family offenses) were excluded as they did not adhere to the objectives of the FERMI project – to study the ramifications of D&FN in the field of violent extremism. Non-forcible sex offenses (incest, statutory rape, and failure to register as a sex offender), motor vehicle theft, drug offenses, prostitution, criminal road violations, human trafficking, kidnapping, extortion, and animal cruelty were omitted as well, due to the absence of academic evidence suggesting a linkage between said crimes and D&FN.

All crime data was standardised, in terms of type and date format, ensuring a single unified dataset of past crime incidents emerges. The four crime types were not present in all collected municipalities, as such, the Dynamic Flows Modeler's training on a specific crime varied in terms of the number of municipalities included (see Table 2). Crime occurrences, within these datasets, were transformed from one crime per row to panel data, with one row being one week and the number of crimes, for each type of crime, as the column.

Table 2: Number of municipalities utilised in training the Dynamic Flows Modeler (per crime)

Crime Type	No. of Municipalities Included
Assault	31
Dest./Dam./Vandalism of Property	26
Disorderly Conduct	24
Larceny/Theft	30

2.2.1.2 Disinformation Data

The D&FN used to train the Dynamic Flows Modeler was the collection of NELA-GT. The first edition was published in 2019, provided approximately 800,000 unique articles, the later versions increased to nearly 1.8 million per year. These datasets represent comprehensive coverage of D&FN's spread in the United States during the studied years and are employed throughout the state-of-the-art (SOTA) literature on the subject of D&FN. The articles included in NELA-GT were scaled for veracity, allowing for the selection of observations that not only met the first, but also the second and third pillars of FERMI's D&FN definition.[8] Utilising NELA-GT meets the GA established requirement of acquiring D&FN data from known D&FN-identification-and analysis' services, being part of the Harvard Dataverse collection.[9] NLP was used to classify NELA-GT's articles into FERMI's three topics of interest: public-health related D&FN, and violent right-/left-wing extremism.[10] The primary objective of doing so was to then understand the intensity of the spread, for each given topic, through the years 2018 – 2022. Important to note, as was comprehensively covered in Deliverable 3.1, that NELA-GT's primarily American-centric and English content poses a challenge as, ideally, FERMI tools would be trained with data sourced from Europe. That being said, **the decision to use NELA-GT followed an exhaustive search and evaluation of existing D&FN datasets and the potential to create new ones**. As outlined in subsection 2.4, the use of NELA-GT for developing the Dynamic Flows Modeler's first and end-product versions, does not prevent future adaptation of the Modeler via re-training with suited datasets that emerge in the future.

Table 3: Considered and excluded D&FN sources for training

Source Name	Reason for Exclusion
CNN / Daily Mail Complied Dataset	Failed to meet project definition of D&FN
LOCO	Failed to meet project definition of D&FN
IRMA	Failed to meet project definition of D&FN and outside languages of interest
Repository of Fake News	Inadequate temporal specification
ISOT Fake News	Insufficient observation count
GermanFakeNC	Insufficient observation count
Spanish Fake and Real News	Insufficient observation count
Spanish Fake News Corpus	Insufficient observation count
GRAFN	Insufficient observation count
FakeCovid Fact-Checked News Dataset	Insufficient observation count
LIAR	Insufficient observation count
Kaggle Fake News Dataset	Insufficient observation count
Albanian Fake News Corpus	Insufficient observation count and outside languages of interest
HoaxItaly	Insufficient temporal range and outside languages of interest
Fakeddit	No geolocation and failed to meet project definition of D&FN
Fake News Dataset	No temporal specification
WELFake Dataset	No temporal specification
FNC-1	No temporal specification
Snopes Fact-News Data	No temporal specification
FakeNewsNet	Requires extensive X API access
COVID-19 Disinfo Dataset	Requires extensive X API access

[8] Ibid.
[9] Ibid.
[10] Ibid.

Before classification could begin, the articles' texts were extracted using SQLite Studio client and all texts underwent a standard NLP pre-processing using Python's natural language toolkit's library, which involved tokenisation and lemmatisation, as well as the removal of stop words (using the toolkit's provided stop words) and punctuation. This ensured that the text data was clean and standardised for analysis.

For COVID-19, the classification relied on a list of keywords provided by the authors behind NELA-GT. The provided list consisted of 241 words, which were then refined to 131, filtering on the basis of relevancy. For political extremism, instead, articles were through manual classification. A choice taken due to the absence of a sufficient keyword list present in academic literature. Manual classification followed closely with the Bundesamt für Verfassungsschutz's (Germany's domestic intelligence service's) definitions, as well as the conceptualisation of extremism in past academic works.[11] Attempts to identify and train the model with left-wing extremism, which were undertaken as laid out in D3.1, did not prove to inform the model differently from training with right-wing extremism. As such, the decision was made to cluster left- and right-wing extremist D&FN.

For additional information regarding the NLP techniques employed in use-case labelling the D&FN articles in NELA-GT, as well as the decision to adopt NELA-GT, please refer to Deliverable 3.1. That being said, intensity was calculated for COVID-19 and right-wing extremist D&FN by considering the number of articles for each respective topic on each day. Intensity was created at a daily level as to allow for varying levels of aggregation depending on the aggregation of crime data. Figures 3 and 4 report their intensities for the years 2020 – 2022. The current iteration of the Dynamic Flows Modeler was trained on data from the aforementioned period, however, as will be explained in Section 5, the Modeler can be subject to retraining with updated datasets to ensure the most contemporary understanding of the phenomenon if and when newer, apt datasets become available. The intensity of D&FN was provided to the model with a one period (a week) lag, to ensure the model understood the direction of the relationship between D&FN and crime.

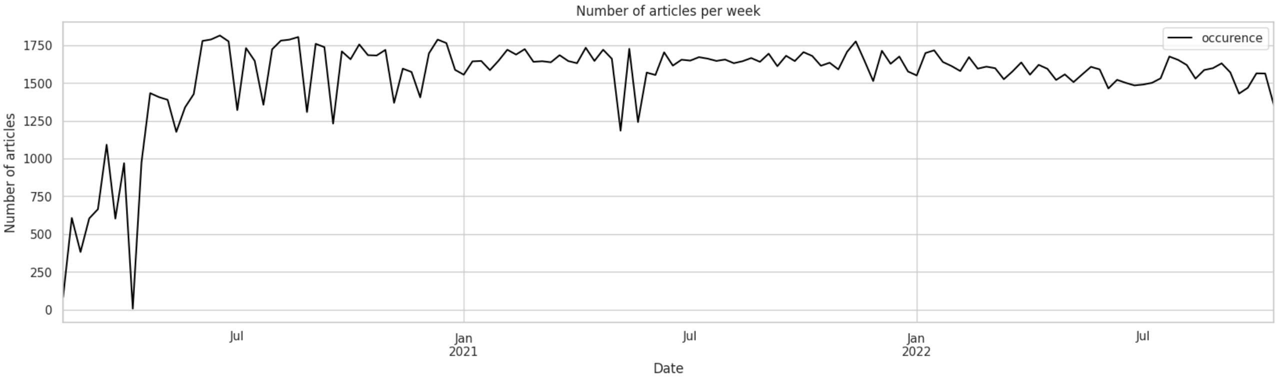
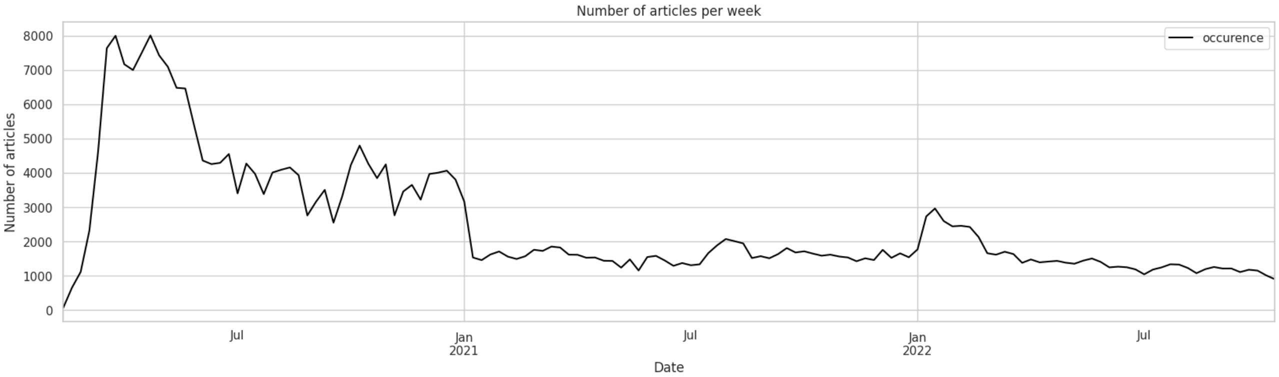


Figure 3: Political Extremism D&FN intensity extracted from NELA-GT (2020 – 2022)



[11] 'Right-wing Extremism,' *Bundesamt für Verfassungsschutz*, n.d.; Torregrossa, J., et al., 'A Survey on Extremism Analysis using Natural Language Processing: Definitions, Literature Review, Trends and Challenges,' *Journal of Ambient Intelligence and Humanized Computing*, 2022; Botticher, A., 'Towards Academic Consensus Definitions of Radicalism and Extremism' *Perspective Terror*, 2017.

Figure 4: COVID-19 D&FN intensity extracted from NELA-GT (2020 – 2022)

With the use of NLP, there arises potential ethical concerns, particularly given the FERMI project's focus of D&FN and how the definition of D&FN can affect the model's understanding. The NLP utilised in the development of the Dynamic Flows Modeler was prelabelled by academic authors as being from known D&FN sources (i.e., labelled as D&FN at a source level), avoiding any potential biases from the tokenisation and lammentisation process applied to said articles. This is aligned with overall FERMI platform, which does not identify content as being D&FN, instead, leaving it to the end-user to submit D&FN for analysis. Additionally, should a European dataset of D&FN that satisfies the quantity and quality of data required to train the D&FN become available, the Dynamic Flows Modeler can be retrained.

2.2.1.3 Addressing Potential Biases in the Training Data

As with the training of any AI-based model, their exists the risk for a bias within the data – either from incorrect or unbalanced datasets being used – to impact the result yet not the evaluation metrics. That is, a bias within the training data can skew the results such that accuracy is not impacted when measured but in real-world application, the model could pose incorrect indications. The choice to include data from the widest possible range of municipalities and the necessity placed utilising significantly large datasets of D&FN arised from attempts to ensure biases were mitigated. By selecting municipalities from across the United States, varying economic, social, and cultural realities were appreciated by the model, as well as different policing practices – a reality faced in Europe. Then, accuracy of the model, as will be reported in greater detail in Section 2.3 was evaluated by predicting in one of these given municipalities. Therefore, the model proved to be applicable in a wide range of geo-political and socio-economic settings, providing confidence that biases were not significantly present. Just as well, utilising NELA-GT as the source for D&FN provided us with the largest possible image of D&FN's presence in the training period – while meeting the necessary conditions for the data to be usable in machine/deep learning model development.

2.2.1.4 Pre-Processing

To prepare the data for input into the model, a windowing methodology was employed. This involved segmenting the data into 12-week intervals, with a stride of 12 weeks, ensuring that there was no overlap between consecutive windows. As a result, the input matrix for each model comprises a 12-week data sequence, encompassing details regarding the specific crime category under examination, along with data on D&FN intensity, mobility changes due to COVID-19, and population, corresponding to the respective place each came occurred. To enhance the model's understanding of seasonality within each input window, additional features, such as month-based dummy variables were introduced. For the model to capture diverse crime data scales, all windows were scaled to a consistent range. The windows were split into training and testing sets, with an 80% allocation for training data and a 20% allocation for testing data.

2.2.2 Machine Learning Architecture

CNN is a model proven to be effective in studying time-series data.[12] It typically consists of two fundamental components: the CNN itself, which extracts and filters the relevant features and the fully connected layer, which produces the estimates using said features and their relevance**.** In effect, the CNN, by studying the past, provides weights to the variables it is provided. To introduce non-linearity into the network, rectified linear unit (ReLU) activation functions are employed in each convolutional layer.[13] The convolutional operation in the $i-th$ layer of the$j-th$ set can be represented as follows:

$$Z[i,j] = W[i,j] * X + b[i,j]$$

Equation 4: Convolutional operation of CNN[14]

[12] Belda, S., et al., 'The Short-Term Prediction of Length of Day Using 1D Convolutional Neural Networks (1D CNN),' *Sensors*, 2022.

[13] Abdeljaber, O., et al., '1D Convolutional Neural Networks and Applications: a Survey,' *Mechanical Systems and Signal Processing*, 2021.

[14] Ibid.

$$A[i,j] = ReLU(Z[i,j])$$

Equation 5: Rectified linear unit activation of CNN convolutional operation output[15]

Where $Z[i,j]$ is the convolutional output, a number expressing the role of a feature in the forecast, $W[i,j]$ is the weight as a matrix, for the $i - th$ layer of the $j - th$ set, X is the provided input, and $b[i,j]$ is the bias term meant to offset the activation function.[16] $A[i,j]$ is then the output, after applying ReLU activation.

The fully connected layer consists of four dense (fully connected) layers. These layers are responsible for further feature refinement and dimensionality reduction. The fully connected part of the network leverages the learned features from the convolutional layers to produce the final output, making it a critical component of the entire architecture for tasks such as regression.

$$Z[k] = W[k] \cdot A[k-1] + b[k]$$

Equation 6: Fully connected layer of CNN[17]

$$A[k] = ReLU(Z[k])$$

Equation 7: ReLU activation of CNN fully connected layer output[18]

Where $Z[k]$ is the output of the $k – th$ fully connected layer, $W[k]$ is the weight as a matrix, for the $k - th$ fully connected layer, $A[k-1]$ is the ReLU activation of the previous fully connected layer, $b[k]$ is the bias for the $k – th$ fully connected layer, and $A[k]$ is the output after applying ReLU activation to the output of Equation 6.

The present CNN design moves to reduce the number of filters, while maintaining the 3 convolutional blocks used in the first version of the Dynamic Flows Modeler. Whereas previously the Dynamic Flows Modeler employed an initial set of 500 filters, followed by 250 and then 128 filters in the proceeding blocks, the first set now contains 32 filters, the second 64, and the third 128. These filters apply convolutional operations, enhancing the network's capacity to recognise significant patterns in the data. ReLU operations also remained from the version one architecture, in each layer. A drop-out rate of 0.2 (or, 20%) was also introduced in the end-produce Dynamic Flows Modeler, to prevent over-fitting.[19] As such, the model was trained to minimise the mean squared error loss. To do so, the model continually adjusts the parameters to improve accuracy and lower the discrepancies between estimates and target value. Equation 8 presents how mean square error was calculated, with *n* being the total number of data points, y_i the target value for the *i – th* data point (the real, unseen by the model, observation), and $f(x_i)$ the predicted value produced by the model, for the *i – th* data point.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - f(x_i)\right)^2$$

Equation 8: Mean squared error loss

2.2.2.1 Retraining at Regular Intervals

The Dynamic Flows Modeler has been trained, as previously mentioned, on data from 2020 – 2022. Considering the General Project Review Consolidated Report's recommendation to "focus on constantly

[15] Ibid.

[16] Abdeljaber, O., et al., 'Operating Machine Learning Across Natural Language Processing Techniques for Improvement of Fabricating News Models' *International Journal of Science System Research*, 2020.

[17] Ibid.

[18] Ibid.

[19] Li, Y. et al., "A Survey on Dropout Methods and Experimental Verification in Recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2023.

updated data sets,"[20] however, **as new D&FN, socio-economic, and crime data become available, the models can be retrained to ensure the persistent accuracy and the appreciation in any changes in the nexus between online D&FN and offline criminal behaviour**. This retraining can be performed once the necessary data has become available or when an eventual end-user LEA provides their private, attune data via the SL framework (see Section 4).

2.3 Achievement of KPIs, KRs, and TOTs

For the development of the Dynamic Flows Modeler, there was four key requirements to be met: (1) the selection of the best performing machine learning approach(es) – while keeping said process detailed and human readable; (2) the acquisition of micro-level data on (a) offline crime and (b) the spread of D&FN; (3) the estimation of spatio-temporal evolution of D&FN-enabled offline crime; and (4) producing bigdata-based profiling of authors and victims of D&FN-enabled and -induced offline crime. Emerging from these key requirements are two KRs (KR2.1 and KR3.1), which, taken together, desire the delivery of a cutting-edge mechanism for modelling and predicting the dynamic flows of disinformation.

The KPIs for the Dynamics Flows Modeler are directly related to said KRs. KPI1.2, Verification of threats and risks identified, to be related to D&FN, in >80% of cases, and KPI2.1, Predictive models for dynamic flows of disinformation with verified increase in their accuracy of >50%. As the project progressed, FERMI's technical partners developed internally assigned targets to more ensure technical excellency among, termed as the TOT. The TOTs for the Dynamic Flows Modeler provide more specific accuracy expectations than the GA assigned KPIs and operationalise the second key requirement. The first TOT, as such, requires the Dynamic Flows Modeler to achieve a MAE that deviates from ground truth values by <=12%, during development validation. The second assigns to the development team the requirement to test five or more different machine or deep learning architectures.

Given the intuitive relation between these various indicators (KRs, KPIs, and TOTs), this subsection will be structured as follows, subsection 2.3.1 will address the development of the model, touching on key requirements 1 and 2, and the second TOT on model selection. Proceedingly, subsection 2.3.2 discusses KPIs 1.2 and 2.1, the first TOT on the Modeler's MAE, providing an overview of the achievements of the model in terms of accuracy. Lastly, subsection 2.3.3 provides an understanding of how the Dynamic Flows Modeler fits the expectation laid out in KR2.1 and KR3.1.

2.3.1 Achievements in Development

In terms of how the Dynamic Flows Modeler was to be developed, the GA, and FERMI partners internal targets, established a clear need for a thorough investigation of which machine or deep learning architectures were best suited for the purposes of FERMI. Moreover, these models had to be trained with micro-level data on offline criminal events and D&FN, ensuring whichever architecture was well equipped to perform. The aforementioned is embodied in the first and second key requirements: (1) the selection of the best performing machine learning approach(es) – while keeping said process detailed and human readable and (2) the acquisition of micro-level data on (a) offline crime and (b) the spread of D&FN. The first key requirement is then further operationalised with the following TOT: >5 machine or deep learning models tested during development.

Indeed, during the development of the Dynamics Flows Modeler, **8 different machine and deep learning approaches were tested**: (1) recurrent neural network, (2) 1D-CNN, (3) CNN, (4) neural network, (5) long-short term memory, (6) an altered version of a transformer attention architecture, (7) XGBoost, and (8) Facebook Prophet. The second, fifth, and sixth were the models included in the Dynamic Flows Modeler's first version, though, as explained in Section 2, further refinement led to the decision to utilise only 1D-CNN in the second version.

On the acquisition of micro-level data on actual offline criminal events and the spread of D&FN, the Dynamic Flows Modeler was trained utilising incident level data from 31 municipalities/counties, thus, the **data for actual offline criminal events was at the most granular level possible**, where each observation

[20] 'General Project Review Consolidated Report (HE) - Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2024

was an individual offline crime. Moreover, through the SL framework, LEA partners, within FERMI, pass to the Dynamic Flows Modeler the needed past crime data to make estimates for end-users, a further micro-level data acquisition, while maintaining LEA's obligation to preserve and protect the personal data held within their servers. **With D&FN, likewise, the data acquired through NELA-GT was at an incident level, where each observation was an individual D&FN event**. For specific details on the collected data, please refer to Subsection 3.1 and Deliverable 3.1.

With regards to development, there exists another requirement specified in the GA, that is, to maintain a process of development which can be characterised as detailed and human readable. Beyond Deliverable 3.1 and this deliverable, that articulate the Modeler's development, it is important to note that **the development of the Dynamic Flows Modeler was done so with entirely public and open source data**. As such, its development is transparent and can always be placed on human review.

2.3.2 Achievements in Accuracy

With respect to accuracy, the Dynamic Flows Modeler need meet three metrics, KPI1.2 – verification of threats and risks identified in >80% of cases; KPI2.1 – verified increase in accuracy of >50% when compared to current methods, and the second TOT – an MAE <= 12% deviation from ground truth values. Given that the Modeler is a tool that predicts outcomes before they happen, accuracy has been tested in a development environment, with past values excluded from training (see Subsection 3.1.3 for further details on the train/test split of data). In turn, KPI1.2 is taken to imply that the Dynamic Flows Modeler is accurate in more than 80% of attempted estimates. Here, MAE is an insightful metric, as it provides an understanding of the average level of performance for the model over the total attempts made. KPI1.2 can, therefore, be considered met when the MAE is <=20%. With this threshold met in the first version of the Dynamic Flows Modeler, the second TOT was adopted, encouraging that an even greater accuracy be achieved, MAE of <=12%.

On said TOT, Figure 5 and Table 4 present **the MAE achieved across the four crime types, in both use-cases. Assault, dest./dam./vandalism of property, and larceny/theft all achieve the TOT with room to spare**. Disorderly conduct, unfortunately, sits 1.5% and 1.3% above the desired MAE. That being said, the variation in how disorderly conduct is defined across jurisdictions is potential explanation for the inability to further reduce error while predicting said crime type. Nonetheless, even in the case of disorderly conduct, KPI1.2 is achieved, with an average accuracy of approximately 85%.

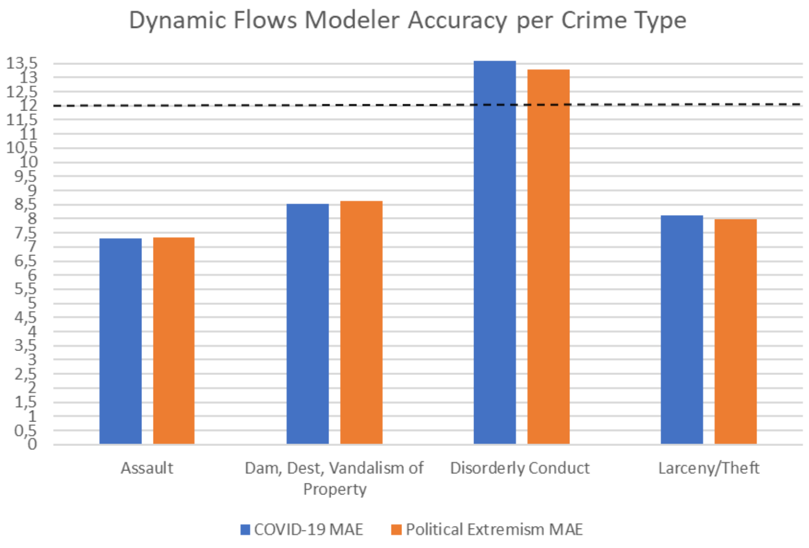


Figure 5: MAE for the Dynamic Flows Modeler across crime types and use cases, graphed

Table 4: MAE for Dynamic Flows Modeler across crime types and use cases

Crime Type	COVID-19 MAE	Political Extremism MAE
Assault	+/- 7.30%	+/- 7.33%
Dam. Dest. Vandalism of Property	+/- 8.54%	+/- 8.63%
Disorderly Conduct	+/- 13.59%	+/- 13.3%
Larceny/Theft	+/- 8.11%	+/- 7.97%

Turning attention towards KPI2.1, verified increase in accuracy of >50% when compared to current methods, operationalization and testing of the KPI was accomplished by providing the same crime data used to train the Dynamic Flows Modeler to a more commonly used statistical method, an autoregressive integrated moving average (ARIMA) model, however, this provision excluding the D&FN spread. Thus, similar to the research discussed in subsection 2.3, the ARIMA can act as a baseline (as currently employed method) in order to gauge the degree to which the Dynamic Flows Modeler improves on said current method. Given the current capacity of end-users, which as detailed in Section 3.3, does not include an existing means of tracking the spread of D&FN, the ARIMA, trained on crime data, represents a method an LEA currently has at their disposal if they so choose to use ML for predictive policing. Therefore, the MAE of said ARIMA, compared to the MAE of the Dynamic Flows Modeler, provides the insight sought by KPI2.1, in terms of the ability of the Modeler to outperform current methods, but **it is worth noting that the Dynamic Flows Modeler, empowered by the Spread Analyser and SL infrastructure, by-and-large stands alone without a comparable technology currently at the disposal of LEAs**.

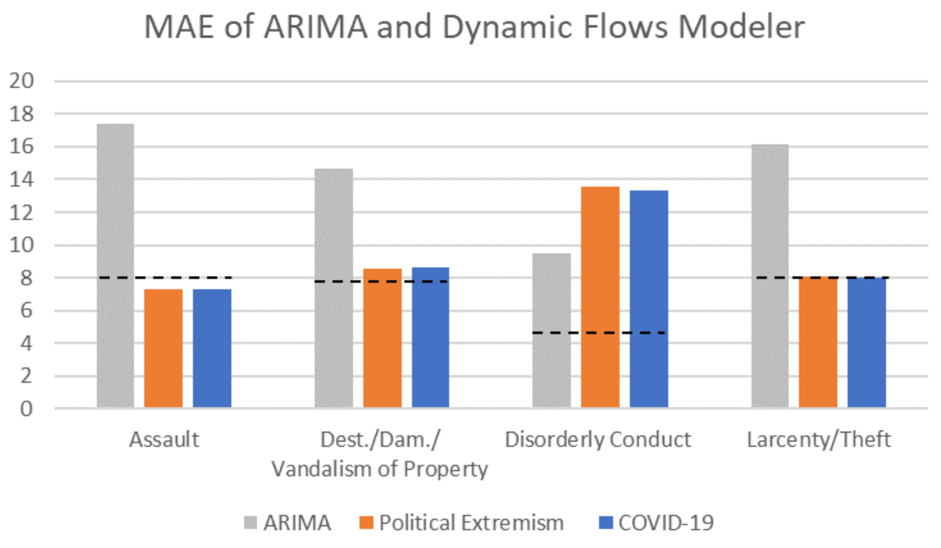


Figure 6: MAE achieved by the Dynamic Flows Modeler and an ARIMA

Figure 6 reports the performance (measured in MAE) of the Dynamic Flows Modeler next to that of the aforementioned ARIMA model. The black line demarcates the 50% threshold established by KPI2.1. Assault and larceny/theft successful meet the KPI, achieving an increase in accuracy 50% or greater compared to the ARIMA, for both use cases. Dam./dest./vandalism of property sits approximately a percentage point above the threshold, however, this still represents a significant improvement on current methods, with an accuracy 41% greater than that of the ARIMA. Again, as is the case with the above TOT (see Figure 5), disorderly conduct is the outlier, here, the Dynamic Flows Modeler had a MAE of 13.59 for extremism and 13.33 for COVID-19. As was alluded to above, the variation in how jurisdictions classify crimes as disorderly conduct is the likely culprit, though the performance of the Dynamic Flows Modeler should be considered, nonetheless, accurate and provides an added layer to the analysis, D&FN, which is absent from the current methods of predictive policing.

2.3.3 Key Results

With respect to T3.1, the FERMI GA calls for the "development of a cutting-edge mechanism modelling and predicting dynamic flows of disinformation"[21] (KR2.1) and, more specifically, the "dynamic flows of disinformation module"[22] (KPI3.1). Taken together, the Dynamic Flows Modeler is the realisation of said KRs, estimating the movement of offline crime following the spread of disinformation offline.

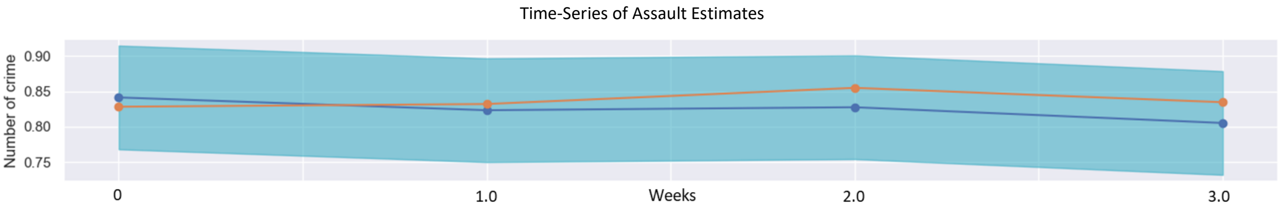
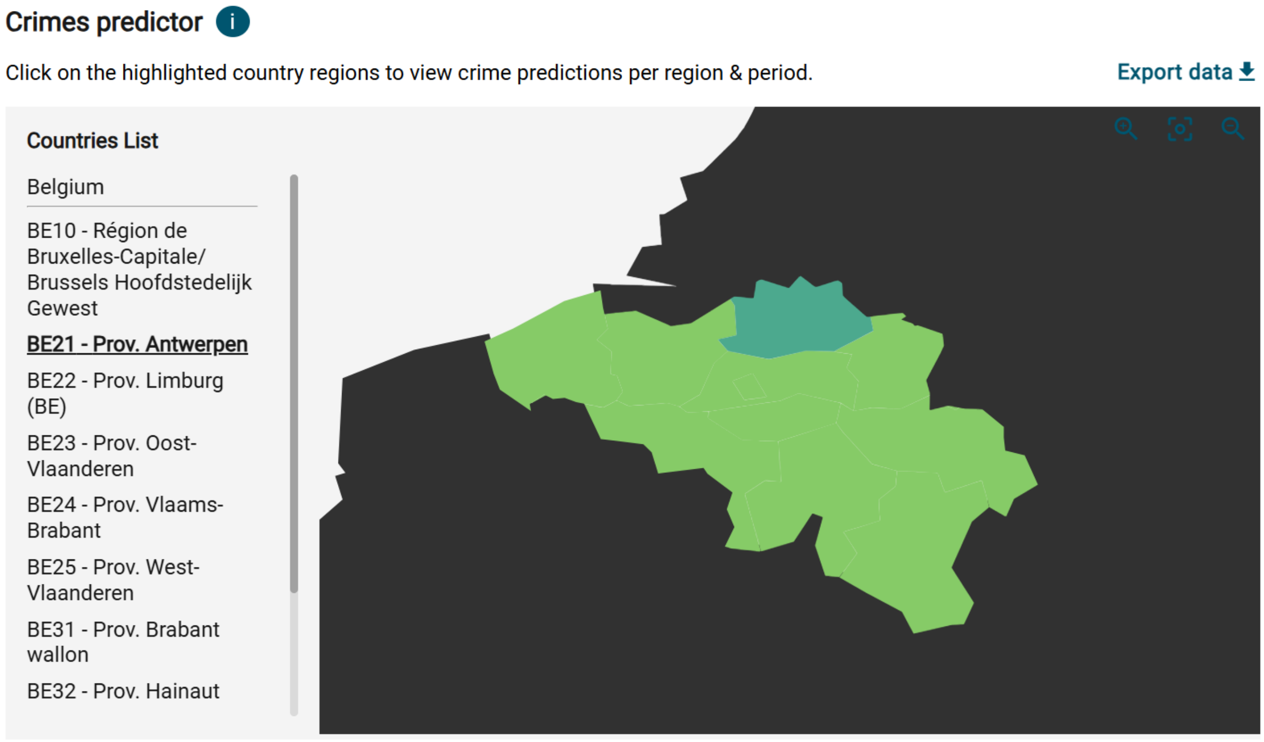


Figure 7: Time-series presentation of an estimated flow of crime, assault

As shown in Figure 7, the Dynamic Flows Modeler provides end-users with estimates for how the four crime types will evolve, or flow, in the weeks following an online D&FN event. Importantly, it does so by leveraging advanced DL algorithms to study and understand the relationship between D&FN and crime. As aforementioned, end-users can gain insight on how crime may potentially flow the 4-week period following the week in which they launch their investigation.



Crimes predictor

Click on the highlighted country regions to view crime predictions per region & period.

Export data

Countries List

Belgium

BE10 - Région de Bruxelles-Capitale/ Brussels Hoofdstedelijk Gewest

BE21 - Prov. Antwerpen

BE22 - Prov. Limburg (BE)

BE23 - Prov. Oost-Vlaanderen

BE24 - Prov. Vlaams-Brabant

BE25 - Prov. West-Vlaanderen

BE31 - Prov. Brabant wallon

BE32 - Prov. Hainaut

Figure 8: Dynamic Flows Modeler in FERMI User-Interface

Utilising the Dynamic Flows Modeler, several research endeavours have been undertaken thus far, specifically with the aim of overcoming the 'black box issue', in turn, advancing the general body of knowledge regarding the relationship between D&FN and offline crime, and understanding the potential profiles of victims/offenders. The black box issue refers to one of the research dilemmas faced when employing

[21] 'Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021.
[22] Ibid.

deep learning algorithms, the depth and plurality of layers relied on while studying the input data makes uncovering the reasoning for the decision (often referred to as explainability) remarkably difficult.[23] As such, two research approaches have been taken, one involving the same data used in training the Dynamic Flows Modeler for end-user use and another where the models were retrained using sample-specific datasets (the architecture was adjusted to adapt to ensure compatibility with the differing data structures). In both approaches, the models produced two sets of forecasts for crime, one without being informed as to the spread of D&FN and one with the spread of D&FN provided. The former acts as a control, while the latter a treatment. The relationship between the two phenomena (online D&FN and offline crime) is then sought in the potential capacity of the spread of D&FN to improve the forecast accuracy for crime, compared to the control model.

The first attempt, as presented at the American Society of Criminology Annual Meeting 2023, found that, using the same data and architecture as the end-user accessible Dynamic Flows Modeler, providing the spread of D&FN improved the capacity of the deep learning algorithms in predicting certain crime types, namely assault and destruction/damage/vandalism of property.[24] A more refined study was then conducted and presented at the 5th International Conference in Electrical Engineering, Information Technology, & Education 2024. In said conference paper, the architecture of the Dynamic Flows Modeler was altered, building on the first presentation, the study was attempted using hate crime occurrences and Google Trends for 5 racially- or politically-charged conspiracy theories (a common form of D&FN spread online), both between the years 2015 – 2019 and in Michigan, United States. The results also showed an increased predictive capacity for deep learning algorithms when informed of the online spread of certain conspiracy theories.[25] A follow up presentation at the American Society of Criminology Annual Meeting 2024, again using a control and treatment model approach, expanded the Michigan-conspiracy theory research to 36 theories, finding further supporting evidence of a crime-D&FN nexus.[26]

Based on the American Society of Criminology Annual Meeting 2024 presentation, a peer-reviewed journal article of the same name was published in the European Journal on Criminal Policy and Research, wherein conclusions regarding the potential victims of conspiracy theory-D&FN are discussed, with reference to the results of the deep learning model with the objective of successfully addressing the key requirement of big-data based profiling of authors and victims of D&FN enabled crime. The research shows that the conspiracy theories able to inform the deep learning models as to the occurrence of offline hate crimes are few, and those few are centred on racially-charged narratives – targeting those of Semitic origins and broader minority/migrant communities present both in Michigan (where the study was conducted) and in Europe.[27] The published paper's methodology and findings were subsequently presented at Eurocrim2025, an academic conference hosted by the European Society of Criminology, with the former being accepted to a session on innovative methods and the latter a session pertaining to information pollution.[28]

2.4 Versatility to Changing End-User Needs

The training of the Dynamic Flows Modeler was undertaken with data from 2020 – 2022, specifically for past crime occurrences and D&FN. It can be contended that while this data scope successfully captures the offline-crime online D&FN nexus, for those years, the relationship between the two phenomena may evolve and

[23] Hassija, V., et al., 'Interpreting Black-Box Models: a Review on Explainable Artificial Intelligence,' *Cognitive Computing*, 2024.

[24] Aziani, A., 'Exploring the Nexus between Information Pollution and Offline Criminal Events,' *American Society of Criminology 78th Annual Meeting*, 2023.

[25] Lo Giudice, M.V. et al., 'Informative (Dis)Information: Exploring the Correlation Between Social Media Disinformation Campaigns and Real-World Criminal Activity,' *2024 5th International Conference in Electronic Engineering, Information Technology & Education*, 2024.

[26] Lo Giudice, M.V., et al., 'Conspiracy to Commit: Information Pollution, Artificial Intelligence, and Real-World Hate Crimes,' *American Society of Criminology 79th Annual Meeting*, 2024.

[27] Aziani, A., et al., 'Conspiracy to Commit: Information Pollution, Artificial Intelligence, and Real-World Hate Crimes,' *European Journal on Criminal Policy Research*, 2025.

[28] Lo Giudice, M.V., et al., 'Disinformation and Crime: The Nexus Between Online Disinformation and Offline Crime,' *Eurocrim2025 – European Society of Criminology*, 2025; Aziani, A. et al., 'Conspiracy to Commit: Information Pollution, Artificial Intelligence, and Real-World Hate Crimes,' *Eurocrim2025 – European Society of Criminology*, 2025.

change as the years progress. Moreover, end-users' needs from the FERMI platform are also subject to change, as social media platforms update and social media users site preferences shift towards new platforms. The technologies prepared for, and currently operational within, the FERMI platform must, therefore, be able to change and remain useful to end-users. The Dynamic Flows Modeler is no exception to this requirement.

The Dynamic Flows Modeler's architecture, as described in Subsection 2.2, is designed to study the relationship between the time-series that have been provided. As time passes from the initial training, the datasets used may become 'dated' and as such, without altering the architectural arrangement, new data can be provided. Indeed, end-user LEAs may choose to provide, via the SL framework, their private data without sharing it, allowing for the Dynamic Flows Modeler to be retrained with contemporary data and ensure it possesses the most contemporary understanding of the relationship between D&FN and offline crime, given that corresponding, apt, D&FN data is available. In the case end-user LEAs do not share their data, even within the privacy protecting framework of the SL framework, FERMI will retrain the Dynamic Flows Modeler at regular intervals, as new data becomes available in a sufficient quantity and quality.

As for changing end-user needs, the FERMI platform currently operates a successful pipeline with X and Mastodon API calls, wherein the Spread Analyser passes to the Dynamic Flows Modeler the level of spread, counted as daily reposts. The applicability of the Dynamic Flows Modeler to other social media platforms rests on this data acquisition being possible. **For any social media platform, the Dynamic Flows Modeler can be applied, so long as there is a measure of spread that is accompanied by date information**, as the Dynamic Flows Modeler requires an understanding of spread that can be transformed into a time-series. Certainly, from a content point of view, reposts is an effective measure, as with each repost, there is the re-circulation of the D&FN contained in the article to potentially an entirely new body of users (i.e., the followers of the account who reposted). Other measures of spread, such as comments (e.g., a high volume of comments from a small quantity of users) or shares (e.g., from a singular account to another, singular, account) do not convey to the Dynamic Flows Modeler, with the same significance, a new body of viewers. However, they can be utilised to inform the model of the spread if a lack of an alternative is present.

In its current end-product version, several end-user needs were incorporated, following feedback received from the first round of FERMI pilots. The user-interface and backend of the DFM were altered, moving away from the first version DFM's focus on the providing users with a summary of model accuracy and towards providing them with an understanding of the distribution of potential offline events, instead. For greater detail on the changes already made to adapt to the pilot-users' feedback, potential end-users' needs, and the results of the second round of piloting, please refer to Subsection 2.1.2.

2.5 Dynamic Flows Modeler Summary

The Dynamic Flows Modeler, through its utilisation of 1-D CNN, represents a noteworthy advancement in the practices of contemporary policing and the capacity of LEAs in responding to D&FN campaigns. Making informed, accurate estimates for the level/quantity of offline crime in NUTS2 regions following online D&FN events, the Dynamic Flows Modeler seeks to assist LEAs in understanding when, where, and how to allocate their resources when dealing with online D&FN. Its application is currently possible with the FERMI use cases' focal points, political extremism (left- and right-wing) and COVID-19/public health.

Between the first (reported in Deliverable 3.1) and second version (reported in the preceding sections), several improvements were made, focusing on streamlining the technological architecture and increasing the accuracy of estimates. As previously mentioned, the second version moves away from a structure that featured several DL algorithms, in favour of a single, best performing, 1-D CNN that predicts for one country at a time. Just as well, it provides results that can be contextualised based on how great a deviation the estimated level of crime is from the level of crime at the time the estimate is being requested (i.e., when the investigation is being made). Moreover, the model is set to be robust to changing end-user needs, in terms of social media platforms and has been show cased with both X and Mastodon API linkages. In terms of KPIs and TOTs provided by the GA and prescribed internally by the technical partners, the Dynamic Flows Modeler does well in accomplishing and achieving said objectives. Likewise, the second version fits the desires of the KRs and key requirements envisioned at the beginning of the FERMI project.

Several changes were also undertaken to the user-interface and backend of the Modeler to better align with end-user needs and the concerns expressed by pilot participants – moving towards a map-based presentation of results, with a colour-coding system to provide users with a clear understanding of the distribution of forecasted changes in event-intensity at a NUTS2 level. Said changes were deemed successful in a second round of piloting, resulting in the Dynamic Flows Modeler achieving all the applicable user-requirements.

In terms of knowledge gained, the Dynamic Flows Modeler, and derivative architectures, has made possible several research endeavours that further our understanding of the nexus between offline crime and online D&FN. Presentations at the American Society of Criminology 2023 and 2024 annual meetings exhibited how providing deep learning models D&FN's spread can better inform them to the occurrence of certain types of crime – suggestive of a relationship between the concepts.[29] The findings were further reinforced and replicated in a presentation (and subsequent publication) at the 5th International Conference in Electrical Engineering, Information Technology, & Education 2024.[30]

[29] Aziani, A., 'Exploring the Nexus between Information Pollution and Offline Criminal Events,' *American Society of Criminology 78th Annual Meeting*, 2023 & Lo Giudice, M.V., et al., 'Conspiracy to Commit: Information Pollution, Artificial Intelligence, and Real-World Hate Crimes,' *American Society of Criminology 79th Annual Meeting*, 2024.

[30] Lo Giudice, M.V. et al., 'Informative (Dis)Information: Exploring the Correlation Between Social Media Disinformation Campaigns and Real-World Criminal Activity,' *2024 5th International Conference in Electronic Engineering, Information Technology & Education*, 2024.

3 Task 3.2 – The Spread Analyser

The Spread Analyser is the pivotal component of the FERMI platform, designed to enhance and launch the investigation process by providing end-users with a sophisticated graph-based representation of D&FN diffusion across social media platforms. This representation is enriched with additional insights derived from extracted data, data analysis, graph analysis, and ML, which collectively support the end-users in understanding the network under investigation. Key features include the identification of bot activity and the assessment of each post's influence, enabling a comprehensive analysis of D&FN spread.

This second phase of development aimed to optimise and enhance the Spread Analyser's capabilities further, including advancements in the graph-building service, exploration of alternative identification methods, optimisation of the graph expansion policy, enhancement of the influence analyser, and improvement of the classification model's F1-Score and accuracy. These improvements have been facilitated through experimentation with various approaches to adopt the most effective solutions. Moreover, the Spread Analyser adheres to legal and ethical constraints outlined in the WP7 deliverables, including the human-in-the-loop approach. End-users are responsible for selecting the social media posts for analysis based on the varying legal frameworks they operate within, ensuring compliance and advancing evidence-gathering. The selection process is informed by the FERMI project's definition of disinformation, focusing on the factual or misleading nature of the information, the intent behind its dissemination, and its potential public harm.

The proceeding subsections are as follows, Section 3.1 gives a practical overview of the Spread Analyser, the feedback it received during the first round of piloting, and how said feedback was incorporated. As such, 3.1 describes the modifications made, delineating the end-product version of the Spread Analyser from the first version detailed in Deliverable 3.1. Proceedingly, 3.2 offers a detailed methodological and technical overview, highlighting compliance with the Grant Agreement. 3.3 addresses the KPIs, KRs, and TOTs, while 3.4 specifies the versatility of the Spread Analyser to changing end-user needs and 3.5 provides a concluding summary of the Spread Analyser tool.

3.1 Practical Description

The goal of the Spread Analyser component is to compile social media data and illustrate the relationship between the post and related account data in the form of a graph structure for further analysis and visualisation. Additionally, it provides insights by analysing the produced graph, focusing on the bot/human classification and the influence scoring the nodes (i.e., posts) of the graph.

Following the second version of development, the component features refined processes and developed new features, further enhancing and expanding the capabilities of the component. This process was guided by the pilot-users' feedback, the next steps outstanding at the time Deliverable 3.1 was submitted, the GA's objectives and the overall goals of the FERMI project and its platform. The aim of the technical effort is to expand the capabilities of the Spread Analyser and further improve the added value to the FERMI platform and the end-users. 3.1.1 focuses upon the developments implemented, and 3.1.2, then addresses the feedback and recommendations received from the Interim Review Report and FERMI pilot.

3.1.1 The Spread Analyser& its Components

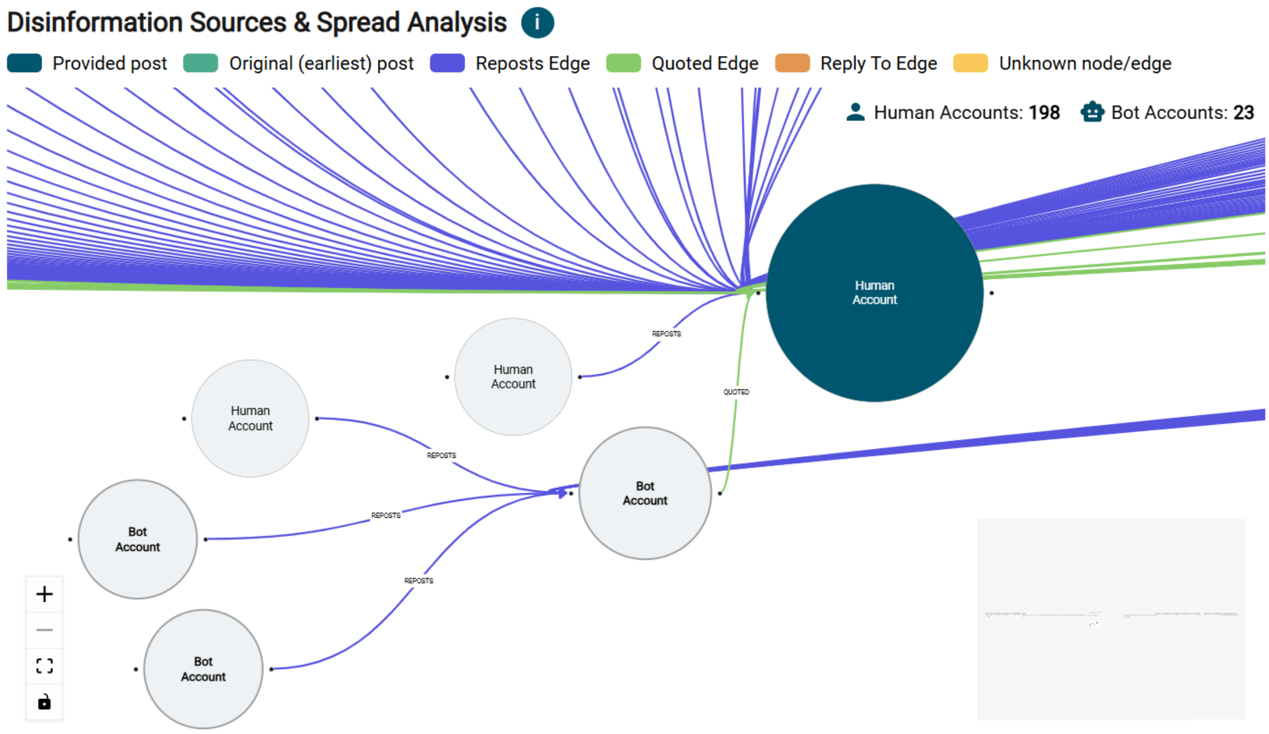


Figure 9: Example network graph of a given D&FN post on social media

The architecture design of the Spread Analyser follows a microservice approach allowing for modular services that interoperate to produce the planned outputs of the Spread Analyser technology offering. The services can be divided into three categories: (1) supporting services, (2) graph building, and (3) graph analysis.

Supporting services include the Orchestrator service that is responsible for managing and monitoring the operational flow of the Spread Analyser component. Additionally, it handles the communication with the rest of the FERMI technology offerings. Furthermore, it includes the neo4j instance used for structuring the graph during building and analysis. Graph building includes the Network graph, and Social media API crawler. These services handle the communication with the supported social media platforms and perform the extraction of information required to build the graph, responsible for building the graph and providing graph related operations (update and stop). Graph analysis is characterised by the services that analyse the produced graph. The analysis task requires parsing the graph data structure, analysing, producing insights and enhancing the graph data structure by incorporating them into the structure. The insights include bot/human classification and influence score calculations per graph node.

To support the incorporation of additional social media platforms the architecture of **the service was updated to facilitate the incorporation of additional social media platforms including Mastodon**, which was incorporated into the supported social media platforms. The new architecture makes the incorporation of social media platforms easier by fully decoupling the API specificities of each social media platform from the graph building logic. Additionally, a new service was incorporated to allow for data persistence of the post analysis requests by the user. As a result, the analysis request is being safeguarded against container failures that could cause data loss.

3.1.1.1 Social Media API Crawler

The social media API crawler is a service that is responsible for effectively using the official API of a social media network to extract data related to the analysis requested by the user. Additionally, it transforms the data extracted in a structure that can be used by the graph building operations (to be described in the proceeding section). Moreover, it is responsible for enforcing the limits set by the social media network's API policies.

The social media API crawler had limited incorporation with the main graph building operations of the Spread Analyser component. To add modular capabilities to the component and incorporate other social media APIs, such as Mastodon, into the component the two services had to be clearly separated. As a result, the X API agent was developed incorporating the functionality of the social media API crawler and adapting it to the modular version of the graph building operations. Additionally, the Mastodon agent was developed in a similar manner, to be operational with the Spread Analyser's graph building operations. Both agents have been configured to be compatible with graph building functionality. Simultaneously, both agents adhere to their respective social media API access requirements, structure and limitations. As a result, the Spread Analyser's social media analysis is adaptable to any social media platform offering access with limited development effort for creating a new agent.

A significant aim of the Spread Analyser and the FERMI platform is to support additional social media platforms. To that end, two distinct actions were taken. The first one, as described above, was to decouple the graph building operations from the social media API crawler. The effect of this process was to significantly reduce the complexity of supporting additional social media platforms. The second action taken, was developing the codebase to support the analysis of Mastodon posts. Conducting research like the one performed in the first phase of the Spread Analyser was required in order to assess which social media platform would be eligible for inclusion. This is important because while there are many social media platforms available only a small fraction of them offer the level of access required to build a graph that would satisfy the data requirements needed to perform graph building and analysis. Amongst the most prevalent limitations were **(1)** full access with commercial account, **(2)** full or limited access with research account, and **(3)** full or limited access with extraction limits.

Full access with commercial account: This is only applicable to users and content interacting with a business account owned by the developers. It is out of scope for the FERMI project case, as the users need to assess information related to third-party accounts.

Full or limited access with research account: Some platforms offer different levels of access to data, provided that, the developers are members of a research organisation. Excluding the platforms that do not offer enough data for analysis, the rest of the platforms can be potentially used leveraging the DSA (Digital Services Act). Two main concerns were presented when pondering this option: **(1)** few of the available platforms would provide access to their data through the DSA but only in a closed "quarantined" environment, hindering data extraction. This would effectively make it impossible to build network graphs. Data could be retrieved and viewed only in a closed webpage offered by the social media platform. **(2)** The DSA requires that only research institutions may have access to the data extracted. This is an issue since the partners developing the Spread Analyser technology offering is a profit-oriented industrial entity.

Full or limited access with extraction limits: The rest of the available platforms are offering access to their data but enforce extraction limits. From the list of social media providers that fall into this category only Mastodon was evaluated as a platform that does offer enough data to build a graph satisfying the needs of the Spread Analyser and the FERMI platform. Two categories of limitations are enforced: **(1)** per account, all user account endpoints and methods can be called 300 times within 5 minutes, and **(2)** per IP, all endpoints and methods can be called 300 times within 5 minutes.

As found during the first phase of the development, almost all social media platforms have enforced significant restrictions on data access. Mastodon proved to be a viable choice to use in the context of the FERMI project. This fact is supported by the increased availability of data extraction methodologies provided by the official Mastodon API. Additionally, the specific social media platform has been expanding its presence with significant increase in its user base, close to achieving ten million users.[31]

[31] 'Number of registered Mastodon users worldwide as of March 2023', Statistica, 2023. https://www.statista.com/statistics/1376022/global-registered-mastodon-users/

3.1.1.2 Graph Building Service

The graph building service is responsible for structuring the extracted social media data into a graph and service is responsible for handling the communication of the graph builder with the rest of the Spread Analyser's services. The graph is built by ingesting the data extracted by the social media API crawler service and processing them into a graph structure while enforcing the build parameters offered by the end-user.

The first version of the Spread Analyser implemented the graph building operations with limited integration with the agent communicating with the X API services. This allowed the incorporation of additional social media platforms but with high complexity. In the second version of the Spread Analyser component the agent for X has been fully decoupled from the graph building operations of the component. The graph building operations were adapted to be agent agnostic allowing for any compatible agent to be utilised with minimized development effort. The result of this process, is the facilitation of new social media platforms API incorporation. Additionally, it provides easy maintenance and allows for functionality extensions.

The service has been robustly developed during the first phase of development with particular emphasis on the structure of data extracted from the official X (formerly Twitter) API. The second development phase was focused on two distinct areas of the Network Graph service functionality: **(1)** enhancement of functionalities and **(2)** graph building operations decoupling from the X agent, adapting the graph building operations to be social media agnostic.

The second development phase **extended on the functionalities of the Spread Analyser component to offer additional configuration and control options to the user**. Firstly, the user is now able to cancel the graph building process while in progress. This functionality serves end-users in cases where the specifications were not correctly set, allowing them to re-start without having to continue a un-needed and technically costly operation. Should new developments be present for a D&FN post that was previously investigated, the end-user is now able to restart the investigation with the same parameters, in other words, there is now an update function for existing graphs. Allowing for investigations to be updated without the need for the end-user to resubmit all the information or generate a new investigation id in the FERMI platform database. In the first development phase the depth and breadth of the graph building were static variables, pre-defined during deployment of the component. In the second version of the Spread Analyser component, these parameters will be configurable by the end-user via the UI. The **breadth and depth parameters offer extensive functionality expansions to the end-user allowing for investigations to be implemented based on the operational goals** of the end users. Below Table 5 reports the parameters' effects on the investigation process.

Table 5: Depth & Breadth Parameters' Effects

Parameters	Investigation effects	Investigation disadvantages
Depth	Delve deeper based on influence, more probable to identify original post	Lose spread, going further in the past but not wide
Breadth	Wider search, based on relationships, better grasp of the D&FN spread	Lose origin post, going wide but not necessarily much further in the past

It should be noted that the time to completion for the graph building is heavily influenced by the required number of requests to the given social media platform's API, exogenous to the FERMI platform. This is a fact that, as explained in Deliverable 3.1, is dependent on the policies enforced by a given social media platform regarding the use of their official API. As a result, high parameter values for depth and breadth will significantly increase the time to completion. Additionally, caps for the total number of monthly requests are enforced in certain API services. This means that if an investigation consumes a large number of the monthly calls, it will significantly reduce the feasible investigations for the rest of the month. To that end, a cap on the value of the breadth and depth parameters will be enforced by the Spread Analyser component.

The additional functionalities developed following the submission of Deliverable 3.1, and described here, aim to offer the user more options to control and configure the investigation process. As referenced in

Deliverable 3.1, these enhancements improve the graph building service, allowing the user to better utilise Spread Analyser's capabilities and influence the graph expansion policy.

3.1.1.3 Insights Extractor

The insights extractor is an intracomponent system that functions as a handler for the services and models enhancing the investigation graph. Additionally, it consumes and makes API requests, within the component, to receive graphs and share their updated versions following the investigation enrichment services application.

3.1.1.4 Influence Analyser

The influence analyser is tasked with scrutinising the graph derived from the initial D&FN input. By leveraging the Graph Data Science (GDS) Library of Neo4J, the influence analyser ranks the graph's nodes from most to least influential. This process aligns with the GA requirement to ensure that "for graph data, graph clustering and graph machine learning algorithms will be developed to detect highly influential nodes propagating disinformation."[32] For identifying the most influential nodes, the analyser employs centrality algorithms alongside the PageRank algorithm from the Neo4J GDS Library. Initially, centrality algorithms were employed alongside the PageRank algorithm. However, in this development phase, a more advanced version of PageRank, the Weighted PageRank, has been adopted. This version incorporates weights between node connections, allowing the influence index to be custom parameterized through these additions, thereby enhancing the accuracy and specificity of each investigation.

3.1.1.5 Bot Modeler

The bot model is designed to determine whether a specific node in the graph represents a human user or a bot account. To develop this system, **deep learning techniques were utilised, specifically an artificial neural network** that classifies each node as either a bot or a human. This approach fulfils the GA requirement, which mandates that "the tool will be able to classify these accounts as physical persons or bots and it will offer for every account an influence index in order to understand their power over the network."[33] Moreover, "for datasets containing ground truth labels, advanced Deep Learning techniques tailored to Natural Language Processing will be employed, in particular the attention mechanism will be combined with recurrent deep networks."[34]

The previous model in use achieves this classification by analysing metadata generated during the graph creation process. In this second phase of development, an initial attempt has been made towards implementing recurrent neural network (RNN) models trained on text data obtained from the metadata of the open-source dataset in an attempt to train the first model. The performance with this NLP-focused solution was not quite satisfactory. A hybrid model has thus been developed. This model although has improved the classification efficiency of the previous RNN by combining text data with other retrieved parameters from the social media platforms is not able to surpass the efficiency of a multilayer perceptron (MLP) neural network dependent on the variables gathered from the social media platforms. Although this hybrid approach is ready for deployment, it currently does not surpass the performance or speed of the existing MLP model. As a result, implementing it may slow down the overall solution without providing significant enhancements.

3.1.1.6 Orchestrator

The Spread Analyser utilises a microservice architecture. To ensure that all services are operating in unison a service is required to unify the different operations into one entity that facilitates the communication with external services. Towards this aim, the Orchestrator service was built and tasked with handling the communication with the rest of the FERMI platform. Additionally, it manages the order in which user-requested analyses are performed.

[32] 'Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021.

[33] Ibid.

[34] Ibid.

The Orchestrator has been operating as the controller of all the Spread Analyser components in the first version of the tool. The second version extends on the same goal by adding fault tolerance methods and updates on the flow of operations. The Orchestrator's service second development period was focused on ensuring that the service gracefully handles errors and supports the update and stop graph functionalities. Additionally, the operations flow management was refined and updated to adapt to the newly incorporated Mastodon social media platform.

3.1.1.7 Enhancements from First Version

In Deliverable 3.1, five 'next steps' were identified and these steps guided the developments undertaken since its submission. The steps were as follows: **(1)** improving the graph building service, including enabling "the user […] to update the graph without the need to initiate a new investigation request; **(2)** trailing other identification methods; **(3)** optimising the adopted graph expansion policy; **(4)** enhancing the influence analyser, and **(5)** improving the F1-Score and accuracy of the classification model.

With respect to the first, the graph building service was improved through the implementation of additional capabilities offering to the end-user additional functionalities. Also, the human in the loop approach was further enhanced by actively allowing the end-user to influence the parameters of the graph building process. The Spread Analyser in its final version supports stopping the graph building or analysis process and the option to re-run a previous analysis to generate a new graph with the same parameters. The end-user is able to choose the depth and breadth of the graph. This way, end users have direct influence on the size and the building flow of the graph. Furthermore, the building service has been significantly decoupled from the first design reported in Deliverable 3.1. Consequently, the incorporation of additional social media platforms is being facilitated.

The Spread Analyser offering was extended to support the analysis of social media posts from the Mastodon platform. This additional feature broadens the range of D&FN sources that the platform can analyze and successfully addresses the second step. The third, optimising the adopted graph expansion policy, instead, was acted upon through enhancing the expansion policy to provide end-users with greater control over the graph-building process. Specifically, users can configure the depth and breadth of the graph. This allows users to influence the size and time to analysis completion. Additionally, **a ranking system was developed for each social media platform to determine which posts should be expanded in each iteration based on the volume and type of relationships to other posts**. On X, quote relationships are considered more important than other types, as quotes provide additional information beyond the original post and further disperse the topic of the quoted post into the network. Similarly, in Mastodon, boosted posts are considered more important because they propagate the topic of the reposted content.

In order to enhance the influence analyser module, bringing the already successfully deployed PageRank solution one step further, a specific alternative implementation of the PageRank was developed. In order to fit the algorithm better to the problem at hands, information gathered through the social media APIs was taken for granted and, specifically, each post's likes that are integrated into the calculation the PageRank performs. **The weighted PageRank calculates the influence of each node in the graph by also looking the popularity of each post and not only based on the original PageRank algorithm, thus enhances and better fits the solution to project needs.**

The fifth step involved improving the F1-Score and accuracy of the classification model used to identify bot accounts. Despite the first version already achieving very good results and performance, the second round of development explored alternatives. The main models involved was RNN models capable to distinguish patterns in the NLP domain. More specifically, long short-term memory (LSTM) was employed and two main alternatives were attempted. The first one was to detect the bots through tweets and by processing only the text and the second one was a hybrid model which combined the texts and the variables gathered from the social media platforms. Both the models developed were not able to surpass the MLPs performance and F1-score achieved. The hybrid model was the best of the two developed in the second phase of development achieving performances similar to the MLP model which makes it capable of being deployed but due to the model's complex architecture and its slower inference it is better suited to project's needs to sustain and in future enhance the MLP model already deployed.

3.1.2 Pilot Feedback and D3.1's Outstanding Steps

As explained in Deliverable 5.2 and Deliverable 5.3, end-user feedback from both rounds of piloting results in overwhelmingly positive evaluations of the Spread Analyser. All KPIs were met in both rounds and few suggestions as to how the tool might be improved were made by the participants. Several other comments were provided; however, they either did not require any platform modifications such as the demand to pick social media posts where more bots are identified, concerning the selection of such posts but not the tool to analyse them (accordingly, this request did inform the implementation of the second iteration of the first pilot, though), or the use of Swedish input data, which is feasible but not within the pilot's purview or happen to be irreconcilable with legal requirements FERMI is bound to uphold.

For instance, the FERMI platform needs to preclude the request to broadly detect "messages of concern" due to the legal and ethical limitations within which the project is operating that happen to ban indiscriminate data analysis (see Deliverable 2.1). Other recommendations ("better labels and descriptions is needed," "better visualization") broadly indicate that some participants would appreciate more guidance and explanations when using the Spread Analyser. The visualisation recommendation was addressed by refining and extending the training material on the Spread Analyser. Furthermore, during the pilot training the technical team supported the process by offering any clarifications required.

That being said, a pilot-user expressed a desire to learn more about the bot detector's reliability, which is relatively clear. Just as well, there was a request to expand the platform's reach, so that it does not remain limited to the analysis of X data but can be fed with posts from other social media platforms. As explained in Deliverable 1.2, this demand is fully in line with the review report the consortium received following the Mid-Term Review. Deliverable 3.1 also establishes that the analysis of further social media data is a primary objective. As explained above, all components developed with the FERMI project have been adapted such that they may be adjusted and applied to any social media platform, should access become available. Moreover, the end-product versions of the technologies have been successfully employed with both X and Mastodon's API.

Moreover, bugs regarding the user inputting posts with no relationships or relationships outside the limitations of the X platform were reported. Accordingly, the Graph Builder and Orchestrator were adapted to discard investigations with only one node (post offered by the end user) and proceed to the next investigation. Just as well, **updates were implemented to provide the enhancements described in Deliverable 3.1**. The end-users have the capability to stop an active post analysis and restart a post analysis request they have previously made. Moreover, end-users can directly affect the building policy of the graph building process, allowing them in this way, to effectively decide on how the analysis they requested will be performed.

3.2 Technical Description

Section 3.2 expands on the services described in 3.1, specifically regarding their advancements following the submission of Deliverable 3.1. The proceeding subsections each cover one service within the Spread Analyser. Overall, the design of the first Spread Analyser version has been expanded with new services to allow for the incorporation of the Mastodon API and the capability to add additional social media platforms with reduced impact to the existing codebase. This is achieved by creating an interface which based on the choice of platform that the user makes, will call upon the associated class object that corresponds to platform specific API implementation. Following this, the class object will be used to extract and process data while building the graph structure.

3.2.1 Network Graph

The Network Graph component has a pivotal role in the offerings of the Spread Analyser. As a service it builds the data graph used for the investigation analysis. The previous section described, in practical terms, the development actions taken. This section, instead, will provide the technical underpinnings of said developments and how they align with the GA's expectations.

Beginning with the decoupling and end-to-end modularity of the Spread Analyser, the process had to be carried out meticulously to ensure that the graph building operations was unaffected. The X agent is operational and that new agents will be integrated into the component without any additional modular implementation required. To achieve this goal all functions that had joined code from the Network graph and the social media API crawler services were decoupled and incorporated into the best suited service. Additionally, **the structure of Spread Analyser codebase was modified to allow for modularity and the incorporation of any compatible agent**. To achieve this a new class architecture was developed. As a result, the social media agents and the graph building graph building operations were interacting through an interface that allowed the application of the agent functions into the graph building operations.

3.2.2 Social Media API Crawler

The Social Media API Crawler is service is comprised of agents that handle the social media platform communication and data collection. Each agent is developed to be fully adapted to only one social media platform. The agents are structured in functions that can be applied to the component graph building operations through an interface class, allowing in that way the application of the specific agents' functionality on the Spread Analyser's graph building process. The agent is also responsible for adhering to the data collection policies of the social media platforms. It also enforces the limitations introduced by the social media platform into the graph building operations. Finally, it performs the necessary data conversions to structure the collected data in a format that can be used by the graph builder service. It should be noted that the content of the data collected is not modified during this process.

3.2.3 Insights Extractor

The insights extractor enhances graphs using the influence analyser and bot classifier model, then returns the updated graphs for further processing by the orchestrator/controller service system. More technical details are going to be described for each module the insights extractor is comprised from.

3.2.4 Influence Analyser

The influence analyser service identifies the most influential nodes within a graph, utilising Neo4J's Graph Data Science library as detailed in the task's GA description. The module focuses on developing machine learning algorithms to detect nodes that significantly spread disinformation, leveraging and extending Neo4J libraries for efficient centrality calculations. Betweenness centrality and PageRank algorithms were explored, with PageRank proving to be more accurate and logically consistent. Consequently, PageRank was selected for the initial version of the influence analyser. This algorithm assesses the importance of each node based on the number and significance of incoming relationships, which can be adjusted according to the specific analysis requirements. For the second version of the influence analysis module, weighted PageRank has been implemented which introduces different relationship weights in the connecting node edges. This is a refinement of the traditional PageRank algorithm which emphasizes the varying significance of the relationships. The underlying mathematical formula for the PageRank version with the weight's implementation is provided in the accompanying equation.

$$WPR(a) = \frac{(1-d)}{N} + d \sum_{x \in N(a)} w(x,a) \cdot WPR(x) \sum_{j \in N(x)} w(x,j)$$

Equation 9: Updated PageRank formula

The traditional PageRank equation calculates the rank of a node by equally distributing the rank contributions from all linking nodes, divided by their outgoing links. In contrast, the Weighted PageRank equation introduces weights to account for the varying importance of different links, normalising these contributions by the total weight of the outgoing links from each page. This allows the Weighted PageRank to provide a more meaningful and accurate calculation of page importance by considering both the quantity and quality of incoming links. **By this alteration the already developed module gains additional value, increasing the accuracy of the total solution**.

3.2.5 Bot Model

As explained above, the bot classification model is designed to determine whether a social media post is authored by a human or not. The model employs deep learning techniques, specifically an artificial neural network, to achieve this goal. Due to the lack of specific bot or human labels in the gathered graph network data, an open-source dataset labelled with this information was used for training. Both the training dataset and the graph network data were harmonised to have the same features to apply the model effectively. Key features of the dataset include user description, follower count, following count, geolocation, language, location description, average social media posts per day, account age, and account type (bot or human).

Textual features were converted into binary variables, while continuous numerical features were normalised to optimise the neural network's performance. The model architecture consists of MLP with two hidden layers containing 128 and 64 neurons, respectively. In the second phase of development, a novel model is in consideration that has the aim of improving the bot detection precision potentially by taking a hybrid approach. The preliminary attempt at employing only text-based models such as RNN has been unsuccessful.

RNNs are particularly useful in NLP tasks, as they capture patterns in sequential data, making them suitable for analysing text. However, for the task of bot detection, a hybrid model that combines text input with the parameters retrieved from the social media platforms has been developed. This approach aimed to leverage the strengths of both text-based analysis and additional metadata to enhance classification accuracy. More complex RNN variants, such as LSTM networks have been experimented with as part of this hybrid model. The architecture of the developed hybrid model consisted of two parallel input branches: one for processing text data and another for numerical features. The text branch utilized an embedding layer followed by bidirectional LSTM layers to capture contextual information in both directions of the text sequence. Specifically, the model employed a 100-dimensional embedding space, followed by a bidirectional LSTM with 128 units returning sequences, and another bidirectional LSTM with 64 units. Dropout layers (30%) were incorporated between LSTM layers to prevent overfitting. The numerical features branch processed account metadata such as follower counts, friend counts, and status counts through batch normalization and dense layers with ReLU activation.

These features were processed through a 64-unit dense layer with dropout regularization. The outputs from both branches were then concatenated and passed through additional dense layers with batch normalization and dropout, culminating in a sigmoid activation function for binary classification. Despite the theoretical advantages of this sophisticated architecture and the additional information provided by the text data, the hybrid model did not exceed the performance of the previously implemented MLP model. The MLP model, which relied solely on numerical features extracted from user accounts, demonstrated superior classification accuracy and generalization capabilities. This suggests that for the specific task of bot detection on this dataset, the behavioural patterns captured by account metadata provide stronger discriminative signals than the textual content produced by the accounts. Therefore, while the hybrid LSTM model represents a more complex approach that incorporates additional data modalities, it will not replace the existing MLP model in the production pipeline. The simpler MLP architecture remains the preferred solution due to its superior performance, computational efficiency, and interpretability. This finding aligns with the principle that more complex models do not always yield better results, particularly when the simpler model has already captured the most relevant patterns for the classification task.

3.2.6 Orchestrator

A new graph building operation was developed for the Orchestrator service to extend the operating capabilities and adapt to the new flow described in the previous section. The updated flow was designed to be social media platform agnostic. This was achieved by enforcing a standardized format for the requests expected by the user and ensuring that the Orchestrator sends the same type of graph building and analysis request to other Spread Analyser services. Additionally, enhancements in error handling were introduced to ensure graceful handling of errors. To safeguard against posts with no relationships available, all graphs with only one post are discarded. Configurations were also implemented to handle requests for stopping graph building and analysis operations respectively. The graph update operation did not require additional configuration since the standardization of the request structure ensures that no additional handling is required.

3.3 Achievement of KPIs, KRs, and TOTs

Section 3.3 will review the KRs, KPIs, and TOTs assigned to the Spread Analyser, explaining both what they entail and how they have been achieved. Specifically, there are two KRs that correspond to the Spread Analyser, KR2.2, the realization of a technology to quickly and accurately analyse D&FN sources and spread, and KR3.2, the development of FERMI's disinformation sources analyser. In terms of KPIs, the GA agreement details the need for the Spread Analyser to achieve 95% accuracy on assessing the origin of D&FN (KPI1.1), an increase in the speed of identifying D&FN sources by at least 60% (KPI2.2) and in accuracy by at least 40% (KPI1.2). Then, in terms of TOTs, TOT1 mandates an F1-score of >81% for the bot model while TOT2 sets a time to achieve result of < 1 minute for the influence analyser.

3.3.1 Key Results

The GA requirement for delivery of the Spread Analyser, characterised in KRs 2.2 and 3.2, can be considered achieved, with the FERMI platform containing a system prototype demonstrated in an operational environment. At this stage, the technology has moved beyond laboratory testing and has also been validated under real-world conditions (i.e, the pilots). The platform has been tested in its intended operational setting and the integration with the existing systems has been ensured. In addition, all the modules have validated their performance metrics, and demonstrated reliability and robustness. The platform has achieved TRL 7 by having showcased that the technology is mature, and has been full-scale deployed. Overall, it is most likely ready for commercialisation, since it has provided confidence to stakeholders about its viability and performance in practical applications. In this respect, the very encouraging and positive end-user feedback on the Spread Analyser provides confidence and confirms the robustness of the end-product.

3.3.2 Key Performance Indicators

With respect to the KPIs, the description provided with the GA refers to the *identification* of D&FN sources and, importantly, benchmarks the achievement of the KPIs to the improvement made relative to the existing capacity of LEAs. In a narrow sense, the Spread Analyser does not identify the sources of D&FN, however, in practice it produces intelligence regarding the most relevant social media accounts present within the network of a given D&FN's spread. Just as well, the Spread Analyser identifies whether the accounts spreading the D&FN are human- or bot-operated, diagnosing the type of accounts driving the given D&FNs spread. Moreover, with the information available to the public and as communicated by FERMI's LEA partners, there is no apparent existing capability held by LEAs to assess the spread and sources of D&FN campaigns, using AI-based monitoring services. As such, the Spread Analyser represents a full-fledged advancement in capacity, as a pioneering technological solution to the emerging threat of D&FN surpassing all relative improvement thresholds stipulated by the above-mentioned KPIs. (Moreover, in the rather likely event the *identification* of D&FN sources alludes to the account spreading D&FN, the Spread Analyser's success rate is even at 100%, considering that all accounts from which the posting and re-posting originates are known to LEA end-users and the spread of their messages is illustrated in the form of the above-mentioned graph.)

3.3.3 Technically Oriented Targets

As aforementioned, two TOTs were to be achieved by the Spread Analyser: TOT1, bot model accuracy of >=81% and TOT2, time to achieve result of < 1 minute with respect to the influence analysis. Beginning with TOT1, F1-score is a metric used in binary problems as the bot classification problem at hand. It incorporates the precision (true positives divided by the total number of positives predicted) and recall (number of true positives predicted divided by the total number of actual positives) metrics.

$$F1 = 2\left(\frac{(percision \ \times recall)}{(percision + recall)}\right)$$

Equation 10: Calculation of F1-Score

In essence, the F1-score is the harmonic mean of precision and recall, symmetrically representing both precision and recall in one metric. The F1-score value can vary from 0 to 1 with 0 indicating the lowest possible value and 1 a perfect precision and recall. F1-score is the most accurate metric that can be used in binary classification and specifically in the analysis of imbalanced datasets. Furthermore, it gives a more objective point of view than just using accuracy as metric which due to not taking the classes into account often leads to misleading outcomes. **The F1-score achieved by the Spread Analyser's bot model is 82%, Likewise, the influence analysis score, produced via the PageRank algorithm, achieves the objective laid out in TOT2, with time to result < 50 seconds.**

3.4 Versatility to Changing End-User Needs

One of the major updates developed and introduced into the Spread Analyser component is the adaptation of the Mastodon social media API. This resulted in the Spread Analyser being able to expand its graph production and analysis capabilities into the Mastodon social media platform. This way the platform's analysis capabilities include X and Mastodon posts allowing for the end user to choose between one of the two platforms posts for analysis.

As a result, the modularity of the component was further refined and its incorporation of additional social media platforms has been made easier and faster to implement. Consequently, **additional social media platforms can be supported, only requiring the development of an agent to connect and use the API services** of said new social media's official API functionalities. The graph building operations of the component has been fully decoupled from the social media platform utilised and thus is not dependent on any one social media.

Following the first pilot phase, the feedback produced by the end users was analysed to extract possible enhancements of the Spread Analyser. The table below lists the analysis results:

Table 6: Pilot feedback identified needs and development actions

Identified need	Implemented feature
Bugs regarding the user inputting posts with no relationships or relationships outside the limitations of the X platform	The Graph Builder and Orchestrator were adapted to discard investigations with only one node (post offered by the end user) and proceed to the next investigation
More social media platforms support was requested	All services were modularised to allow for different social media platforms to be supported. The Mastodon agent was developed and incorporated into the available social media platforms to support investigations.

Furthermore, the influence analysis module is already able to give results regardless the social media platform integrated.

3.5 The Spread Analyser Summary

As the component that begins the FERMI platform's pipeline, the Spread Analyser is designed to launch the investigation process by communicating via API with a given social media platform, provide end-users with a sophisticated, structured graph-based representation of the end-user provided D&FN post's spread, online, and provide insights as to the type and influence of accounts within said spread. Advancing the intelligence picture available to LEA end-users by classifying the accounts within the network as being human- or bot-operated and assigning to each an influence score, telling as to the relevancy the accounts have in the D&FN's diffusion.

The development of the end-product version of the Spread Analyser, in the period following the submission of Deliverable 3.1 optimised and enhanced the Spread Analyser's capabilities, pushing forward the graph-building service, exploring alternative identification methods for bot accounts, and streamlining end-user's interaction, providing a breath and width selection, creating the option to update graphs, and allowing for the cancellation of in-progress investigations when the specifications did not match the end-user's intentions. Pilot-user feedback was appreciated, when ethical and legal constraints allowed, and can overall be described as a positive reception for the first version (which has nonetheless been advanced in the end-product version). The defined and imposed KRs and TOTs were successfully met, while KPIs regarding improvements upon the current capacity of LEAs was addressed, given that the Spread Analyser represents a pioneering technology and, based on publicly available knowledge and information provided by LEA partners within the FERMI consortium, the first of its kind.

4 Task 3.4 – The Swarm Learning Framework

This section details the SL infrastructure (T3.4) and changes/updates introduced to the component in this second development phase. **SL is designed to offer a scalable software architecture that allows machine learning models to be trained directly at the data source, ensuring sensitive information remains secure** and protected by eliminating the need for data transfer. Integrated within the FERMI platform, and working alongside the Dynamic Flows Modeler (T3.1), this technology enables the analysis of historical crime data from various independent LEAs without compromising privacy or data ownership. This approach not only supports privacy compliance but also fosters effective collaboration between LEAs by ensuring data remains securely on their servers, addressing key concerns around sharing sensitive information.

We begin with a practical overview of the tool, followed by an overview of pilot feedback and the steps that remained to be addressed by the time the preceding deliverable, Deliverable 3.1, was submitted, an in-depth technical explanation of how these practical features have been implemented. Additionally, we discuss how the tool meets various KPIs, KRs, and TOTs, and conclude with an explanation of its adaptability to evolving end-user needs.

4.1 Practical Description

SL is one of the tools included within the FERMI platform. In this section, we will describe it from a practical point of view in a summarised way, as it was deeply covered in Deliverable 3.1. This framework enables the training of global models using the private data of independent agents, which are LEAs in the case of FERMI. These LEAs follow strict privacy and data protection regulations, preventing them from sharing sensitive data across jurisdictions. SL serves as a bridge, allowing them to benefit from cross-jurisdictional data analysis without compromising privacy. It is connected with the Dynamics Flows Modeler, which, through SL, can analyse past crime occurrences across Europe without the need for LEAs to transfer confidential data. This provides a unique opportunity for crime forecasting while preserving data integrity and privacy.

The SL framework of the FERMI platform has been developed using the Fleviden tool, an extensible and modular federated learning framework designed by the Research and Development Department at ATOS. This tool provides the necessary technical infrastructure to implement these SL features, enabling a scalable software architecture that perfectly suits the needs for multi-party data analysis without compromising privacy and data protection. It employs a pipes and filters architectural pattern, which **facilitates the addition of new functionalities and the efficient management of data flows**. It includes several classes such as *Client*, *Server* or *Trainer*, which are designed to interact through input and output interfaces, allowing for modularity that supports both client-server communication and advanced privacy protocols. Fleviden's adaptability has enabled it to extend its capabilities to specifically meet the requirements of FERMI, providing a decentralised architecture that ensures compliance with data protection regulations and minimises attack surfaces, while facilitating dynamic and agile collaboration among multiple LEAs.

The key features of the SL framework include:

> **Dynamic Server Selection**: the SL framework employs a dynamic server selection mechanism that rotates the server role among the agents using a round-robin policy. This not only ensures that no single agent can dominate the learning process but also helps in balancing the computational load across the network. This policy is instrumental in fostering trust among participants, crucial for collaborative environments involving sensitive data. In summary, this rotation of server role ensures fairness and robustness in the learning process.
>
> **Local Model Aggregation**: this process involves the collection and synthesis of local models from each agent by the server. The server agent uses advanced algorithms to validate and integrate these models into a comprehensive global model. This step is vital for preserving the quality and accuracy of the model, as it filters out erroneous data inputs, ensuring that the outcomes are robust and reliable. The integrity of this process is central

to the success of the SL framework, as it directly impacts the predictive power and utility of the global model.

Global Model Redistribution: after successful aggregation, the global model is redistributed to all participating agents. This redistribution process is equipped with fault-tolerant communication protocols that ensure each agent receives the updated model despite potential network failures or data transmission errors. Techniques like multiple retries and error logging are used to handle these challenges efficiently, ensuring continuous operation and system resilience.

Scalability: the scalability of the SL framework is facilitated by Fleviden's flexible architecture, which can efficiently manage an increasing workload and a growing number of agents. This capability is crucial for adapting to expanding operational needs without sacrificing performance or speed. Scalability ensures that as more LEAs join the platform, the system remains efficient and responsive, capable of processing large volumes of data and supporting complex computations.

Integration with the Dynamics Flows Modeler: the SL framework is integrated with the Dynamics Flows Modeler, which utilises the aggregated global models to perform sophisticated crime pattern analysis. This integration allows LEAs to leverage collective intelligence without sharing raw data, aligning with strict privacy and security standards. The insights generated by the DFM are instrumental for strategic planning and proactive law enforcement, enhancing public safety with data-driven precision**.**

Privacy-Preserving Mechanisms: to safeguard the privacy of data while enabling valuable insights from aggregated models, the SL framework incorporates state-of-the-art privacy-preserving mechanisms. These include Secure-Sum, which allows data to be summed without exposing individual contributions, and Differential Privacy, which adds Gaussian noise to obscure the specifics of data entries. These techniques ensure that the system adheres to privacy regulations and builds trust among participating agencies by guaranteeing that no sensitive information is compromised.

With all these features, the SL framework within the FERMI platform significantly enhances the ability of law enforcement agencies to cooperate. By enabling secure, private, and efficient data sharing and analysis, it improves LEAs' capabilities to predict and prevent crime, contributing to a safer environment through collaborative intelligence.

4.1.1 Pilot Feedback and D3.1's Outstanding Steps

After this brief overview of the framework, we will now explore how we have integrated feedback from end-users in the first round of pilots to advance the development of this tool. The FERMI platform has undergone extensive testing and improvements, driven by invaluable insights from those end-users who have used it. **The feedback from the first round of pilots highlighted critical improvement areas within the SL framework, particularly the need to ensure privacy and data protection while improving end-user interaction**. Specifically, the Swarm Learning framework sought to improve on UR019 (secure collaboration between LEAs), UR036 (compliance with data protection regulations), and UR031 (availability of accurate information on offline crimes). As reported in Deliverable D5.2, "[w]ith an end-user satisfaction of 38% (target >65%), users showed limited confidence in the FERMI platform's ability to facilitate inter-agency collaboration without data sharing. The low satisfaction score suggests that users may need more robust features for secure and indirect collaboration across agencies, possibly through anonymised or aggregated data-sharing options."[35]

[35] Deliverable 5.2 – FERMI 1st execution reports, p.40.

In response, the Swarm Learning tool has been, and is continually capable of being further, refined to facilitate more secure and efficient collaboration among law enforcement agencies without the need for external data sharing. This includes introducing secure sum protocols to ensure data protection and trust, which allows for the merging of AI model outputs while keeping individual contributions private.

Additionally, **the user interface has been improved to better describe and represent the connections between D&FN and offline crime, making it easier for LEAs to interpret and use the data effectively**. To clearly demonstrate the functionality and benefits of these updates, as well as the overall advantages of the Swarm Learning module, a video demonstration was created, allowing end-users to see the operation of an otherwise back-end component and highlight the Swarm Learning infrastructures capabilities in supporting LEAs in secure and effective collaboration.

Having discussed how we have integrated end-user feedback to refine our tool, we now shift focus to evaluate the progress made regarding the next steps identified in Deliverable 3.1, with most having been achieved. In Deliverable 3.1, it was stated that alignment between the FMI, BPA, and BFP datasets has to be completed, mapping of the crime types present in the different datasets provided by the pilot hosting partners was necessary, and that the socio-economic controls for Germany and Belgium needed to be collected and adapted to the one already sourced from Finland. These points have been covered, and the technology is now able to predict the number of crimes in all the NUTS-2 areas of the three mentioned countries. Once we had all the data, the **AI models were retrained**, so the parameters and hyperparameters of the algorithms were adapted to the new datasets, as it was stated in Deliverable 3.1 ("it is possible that some parameters and hyperparameters of the algorithms must be adapted to extract the best possible model from the available data").

Apart from that, we had in mind that "the output of the swarm learning framework has to be adapted to the Dynamic Flows Modeler". This required adaptation and synchronisation with that component, which have been achieved through various improvements of the tool:

> **Integration Updates**: the SL component has now been deployed as an independent entity. Unlike the previous setup where the crime predictor's output was directly integrated within the Dynamics Flow Modeler, the current implementation allows the Dynamic Flows Modeler to request crime data from the SL component in real time, so the predictions are inferenced by a live component which contains the final model trained using the SL methodology.
>
> **Refined Crime Type Forecasting**. While the SL model continues to predict the same number of crime types, eleven in total, the types of crimes reported have been narrowed down to four: assault, destruction/damage/vandalism of property, disorderly conduct, and larceny/theft. These changes are made to align with the crime types handled by the Dynamic Flows Modeler. Other crime types previously included (homicide, burglary, arson, forcible sex offenses, weapon law violations, intimidation, and trespassing) have been excluded from the output as they are no longer utilised by other components of the platform.
>
> **Extended Prediction Period**. The prediction period has been extended to cover 2023-2025, with the current model now providing data from 2019 through 2023-2025. This update is motivated by the need to support the examination of current disinformation campaigns on the platform, necessitating data availability up to 2025. The extension uses the latest available statistics for the NUTS-2 regions on the Eurostat website, which currently extend up to 2023 for most of the variables.[36] For the years 2024 and 2025, it is assumed that conditions remain similar to 2023, and therefore, certain values have been replicated in the dataset to maintain continuity and relevance of the predictions. Importantly, the replicated data is in place for the piloting stage, allowing for the FERMI platform to be tested by end-users. As the tool shifts from piloting to real end-user

[36] Eurostat. (2019-2025). Eurostat Database. https://ec.europa.eu/eurostat/web/main/data/database

applications, these variables can be updated with the most recent available or with data sourced directly from the end-user.

More information about the updates made to the Swarm Learning framework in this second development phase will be provided in Subsection 4.2.

4.1.1.1 Results of the Second Round of Pilots

As was the case for the Dynamic Flows Modeler, the Swarm Learning framework was evaluated in the pilot 2 session, of the second round of pilots. With 23 total pilot-users (consisting of 7 active-duty LEA personnel, 5 non-active duty LEA staff, 11 LEA advisers, and 2 acquisition experts) and 23 completed questionnaires. The evaluations reflected again the successful adjustment on the part of the FERMI technological offerings to the feedback provided in the first round of pilots**, with 95.65%, 91.30%, and 100% pilot-user satisfaction scores being achieved for UR019, UR031, and UR036**, respectively. Further and more detailed reporting on the results of the second round of pilots can be found in Deliverable 5.3.

4.2 Technical Description

This section covers the methodology and technical details explaining how the SL operates, particularly the latest updates since the first stage of development. Subsection 4.2.1 will explain the updates of the internal components used by the tool, while Subsections 4.2.2, 4.2.3, and 4.2.4 will deeply explain the changes already stated in Subsection 4.1 and 4.1.1. Finally, Section 4.1.5 addresses the specific GA functional requirements relevant to this tool.

4.2.1 Updates in the Fleviden Tool

As it was explained in Deliverable 3.1, the SL has been developed using Fleviden, an extensible framework for FL structured around the pipes and filters pattern, developed by ATOS. Its core functionality revolves around *pods* which manage different aspects of FL, such as aggregation, local training, synchronisation and secure communication. For FERMI, Fleviden has been extended to meet the GA’s requirement of a completely decentralised system that ensures data protection, minimises vulnerabilities, and allows dynamic collaboration among LEAs across Europe.

New pods and classes were introduced to support SL’s specific needs. These classes extend Fleviden’s core functionalities to enable features like the server rotation and privacy-preserving mechanisms. Since the first deployment, new pods have been added to the tool, the packages available in the framework have been modified, and others pods have been slightly changed or augmented their functionality. In the following table, the final pods used in the deployment and the updated descriptions are summarised:

Table 7: SL Framework’s Final Pods

Pod	Package	Functionality
Agent	Architecture	This pod implements a SL agent that can coordinate with other agents to train a global model in a federated learning fashion. The agents involved in the protocol assume a given index as their identities range from zero to the number of agents. This index is used to take over the role of the central server in charge of aggregating the local models from the other agents. The agent who acts as a server is selected in a round robin fashion based on its index.
HTTP	Bridge	This class allows to connect two pods via HTTP. The pod acts as a web server and as a client at the same time. Any wire can be connected to the HTTP pod using the `Pod.link()` method. Afterwards, the connected wire can be bridged via ‘HTTP.bridge()’ by specifying a host, endpoint and port in the target

		machine. In the target machine, `HTTP.wait()`can be used to wait for any HTTP request by specifying an endpoint and an output wire.
Gaussian Noise	Privacy	This pod adds Gaussian noise to model parameters to ensure differential privacy. Adding Gaussian noise in federated learning is essential for enhancing privacy and data protection while maintaining model performance. This technique helps protect individual data by ensuring that locally trained models, when shared, do not reveal sensitive information. By applying Gaussian noise, the system achieves differential privacy, meaning that even if a node is compromised, it becomes difficult to extract specific details from other nodes' data. Additionally, Gaussian noise mitigates the risk of inference attacks, where attackers try to deduce private information from the shared models. Balancing accuracy and privacy, the noise is carefully adjusted to safeguard data while minimising any significant impact on the model's effectiveness.
Secure Sum	Privacy	This pod implements a secure-sum protocol, which ensures that individual model updates from different nodes are encrypted so that the server can only see the aggregated result, preventing it from accessing individual contributions. Each agent generates a private random mask of the parameter vector, which is regenerated in each round to ensure privacy across rounds. This private mask is shared with all the other agents except for the one acting as server in that round. Instead of sharing the entire mask, each agent shares the random seed of a pseudo-random number generator (this represents a constant communication overhead with respect to the number of parameters). With this random seed, each agent can generate the masks of all other agents. Before sending the parameter vector to the agent acting as the server, each agent adds its private mask, ensuring the server is unable to retrieve the actual parameters. After the server computes the sum of all the received parameter vectors, the aggregation is sent to the agents, where they subtract all the private masks obtained from the other clients' seeds. For additional protection against semi-honest participants, differential privacy techniques (adding Gaussian noise) is applied as already explained, further securing the data from potential inference attacks
Keras / PyTorch / Scikit-Learn	Trainers	A Keras / PyTorch / Scikit-learn pod that trains, evaluates, and predicts using a model from one of these libraries. In the use-case at hand, Keras library is being used to train and evaluate the models.
CSV	Loaders	A CSV loader pod loads data from a csv file using the pandas library.
Weighted Aggregator	Aggregators	This pod implements a weighted aggregator mechanism. It contains the logic to aggregate gradients or weights and to apply clip normalisation for the prevention of exploding gradients. When dealing with gradients, a learning rate can be provided to apply a step of stochastic gradient descent (SGD) before aggregating them.
Starter	Flow	This Starter pod triggers the specified interfaces at the beginning of the fleviden cycle. It is used for starting a flow of messages throughout the programme and quick specific actions.
Ender	Flow	The Ender pod allows the user to properly finish the Fleviden cycle when a message is received through a specified wire.
Rotator	Flow	This pod allows to specify a number of wires that gets created and triggered in different stages in a round-robin fashion. It creates a given number of wires named '/send-to-i', where i < num_wires. When a request via the `/send` wire is received, the 'Rotator' pod picks one of the '/send-to-i' wires and forwards the received request through it.
Juncture	Flow	This pod allows to combine several wires into a single one. The juncture registers several input wires and when all of them are triggered, the juncture combines their outputs into a single output interface that is triggered a single

		time. Afterward, the juncture awaits again all the interfaces. This pod is used, for example, to ensure that the data have been properly loaded by the CSV pod before the trainer pod is triggered.
Logger	Debug	A pod can then print and save messages with different levels of criticality.

It must be noted that privacy enhancements, including the addition of the Secure Sum Pod, are key features of this second version of the Swarm Learning tool. These improvements, made in response to feedback from end users, enhance security and efficiency, facilitating collaboration among law enforcement agencies without the need for external data sharing.

4.2.2 Retraining of AI models

As long as the data available changed, the model used was also updated. After testing with different models, architectures and parameters, we are finally using an MLP regression model, with proves to produce the best accuracy. The layers of the model are the following: a dense layer of 128 neurons with ReLU activation; a dropout layer with 0.1 dropout rate; a dense layer of 64 neurons with ReLU activation; a dropout later with 0.1 dropout rate; a dense layer of 32 layers and ReLU activation; and a final dense layer of 1 neuron (that is, the output layer for regression).

The model begins with an input layer that matches the shape of the dataset, which allows flexibility in handling data with various feature dimensions. The initial layer has 128 neurons with a ReLU activation, which helps the network learn complex, non-linear patterns in the data. The model then uses three dense (fully connected) layers, progressively reducing the number of neurons from 128 to 64, and finally to 32. All dense layers employ the ReLU activation function, known for its efficiency in deep networks by mitigating issues with gradient saturation. Reducing the neuron count across layers compresses information, enabling the model to distill key patterns in the input data while retaining the essential predictive power of the initial larger layer.

Two dropout layers with a 0.1 rate are included to enhance the model’s generalisability by randomly deactivating 10% of neurons during training. This dropout serves as a regularisation technique, helping prevent overfitting and ensuring the model can better generalise to new data. The output layer, with a single neuron, provides a continuous output, making this architecture well-suited for regression tasks like the one at hand. With this model, we have reached the target accuracy, as it will be shown in Subsection 4.3.

4.2.3 Integration Decisions: New Inference Service

As stated in Section 2, the SL component has been integrated within the platform, so the Dynamic Flows Modeler component is able to make predictions in real-time. To achieve this, the output of the SL component has been implemented in this second stage through an inference service. The Flask-based inference service implemented enables the storage and use of the last trained model within a Swarm Learning system to predict crime types across different regions. It has two main endpoints:

> **Endpoint /save-model**: this endpoint receives a model in JSON format, sent by the last agent in the Swarm Learning network once the latest global model training is completed. The JSON model is saved to a file, ensuring it is readily available for subsequent predictions. This design guarantees that the system always operates with the most up-to-date, collaboratively trained model.
>
> **Endpoint /predict**: the endpoint loads the stored model and uses it to make predictions based on input data provided in the request body. When a request is sent to /predict, such as with a curl command, the service expects data including the country, and a time range (start and end), specified by week, month, and year. For example:

```
curl -X POST http:// 49.13.163.113:5001/predict -H "Content-Type: application/json" -d '{
  "data": [
    {
      "Country": "Finland",
      "start": {
        "week": 2,
        "month": 1,
        "year": 2019
      },
      "end": {
        "week": 6,
        "month": 2,
        "year": 2019
      }
    }
  ]
}'
```

Figure 10: Example of Valid Request for the New Inference Service

Upon receiving a request like the example for Finland, the service responds with crime predictions for each NUTS2 region within the country, segmented by weeks, for specific crime types expected by the requesting component, the Dynamics Flows Modeler. The output includes: **CrimeCount**, forecasted crime count values for each week; **CrimeType**, type of crime (e.g., assault, theft, Dam./dest./vandalism of property, disorderly conduct); **NUTS2Code**, the specific NUTS2 region code for which the prediction is made, and **Time Information**, lists of the weeks, months, and years covered in the prediction.

```
{
  "predictions": [
    {
      "NUTS2Code": "FI19",
      "CrimeType": "assault",
      "week": [2, 3, 4, 5, 6],
      "month": [1, 1, 1, 2, 2],
      "year": [2019, 2019, 2019, 2019, 2019],
      "CrimeCount": [74, 81, 78, 69, 63]
    },
    {
      "NUTS2Code": "FI19",
      "CrimeType": "destruction, damage, vandalism of property",
      "week": [2, 3, 4, 5, 6],
      "month": [1, 1, 1, 2, 2],
      "year": [2019, 2019, 2019, 2019, 2019],
      "CrimeCount": [67, 73, 71, 62, 56]
    },
    {
      "NUTS2Code": "FI19",
      "CrimeType": "disorderly conduct",
      "week": [2, 3, 4, 5, 6],
      "month": [1, 1, 1, 2, 2],
      "year": [2019, 2019, 2019, 2019, 2019],
      "CrimeCount": [239, 281, 263, 205, 185]
    },
    ...
    {
      "NUTS2Code": "FI1B",
      "CrimeType": "larceny, theft",
      "week": [2, 3, 4, 5, 6],
      "month": [1, 1, 1, 2, 2],
      "year": [2019, 2019, 2019, 2019, 2019],
      "CrimeCount": [51, 57, 54, 46, 40]
    }
  ]
}
```

Figure 11: Example of Output for the Previous Inference Service Request

Each prediction block in the output represents the results for a specific crime type in a particular NUTS2 region, aggregated over the requested period. This design enables flexible and detailed crime forecasting, aiding in dynamic analysis and decision-making based on predicted crime patterns across time and location.

4.2.4 Time Periods to be Covered: Updated Features

When updating the datasets to enable 2023 predictions, it was observed that not all the statistic data required by the model had been fully updated in Eurostat. As a result, the model's features had to be adjusted and / or changed. The following variables, which were fully available for the period between 2019 and 2023 in the three covered countries (Belgium, Finland and Germany), were ultimately selected: density, GDP per inhabitant, low education, median age, population, and unemployment.

These features were chosen for their relevance in crime prediction models, as they significantly influence crime patterns. Density and population affect the concentration of people and the likelihood of interactions, which can influence crime rates. GDP per inhabitant reflects the economic status of a region, correlating with crime levels in both high- and low-income areas. Low education rate and median age relate to socio-economic and demographic factors that may correlate with risk behaviors or vulnerability. Finally, the unemployment rate is a key economic indicator that impacts crime levels, especially for crimes linked to economic opportunities or social pressure.[37]

Apart from adding new features, some were transformed to improve the accuracy of the model. This is the case of the density and the population, which have been logarithmic scaled, mirroring the approach employed in T3.1. The reason behind that is that these variables tend to have wide-ranging values across different NUTS-2 regions. Using this scale reduces the impact of extreme values, making these features more comparable across diverse regions.

With these changes, and keeping unchanged the features related to the period of the year in which the prediction takes place, the final features used by the model are: year, week (sin), week (cos), low education rate, median age, unemployment rate, population density (log), and population (log).

4.2.5 GA Functional Requirements

Regarding the functionalities that SL should include, the GA stated that "a permissioned blockchain network will be used for the onboarding of the nodes or agents that will participate in the framework and for sharing the learnings in a safe and secure manner." In place of the original network for onboarding nodes and securely sharing learnings, the decision was to use Gaussian noise and secure sum due to several practical and performance-related considerations.

Firstly, while blockchain provides a robust method for ensuring immutability and trust in decentralised environments, its deployment, particularly in a permissioned setup, introduces significant complexity. Blockchain networks require extensive configuration, consensus mechanisms, and potentially add substantial computational overhead to ensure each node follows the protocol correctly. This additional processing and validation may slow down the federated learning process, reducing scalability and efficiency, especially in a dynamic environment like SL where nodes act as both clients and servers, and roles frequently change.

Moreover, the integration of blockchain would increase network load, as each node would need to validate transactions or learning updates, leading to latency and higher communication costs. This could make real-time collaboration between nodes difficult to achieve. Additionally, permissioned blockchains introduce

[37] Wang, Y., Klabjan, D., Pei, J., 'Prediction of crime occurrence from multi-modal data using deep learning,' *PLoS ONE*, 2017.

maintenance challenges, such as managing the onboarding process, ensuring proper permissions, and handling potential forks or discrepancies within the network.

In contrast, **using Gaussian noise and secure sum offers a lighter and more scalable solution**. Gaussian noise provides differential privacy, ensuring that individual contributions remain anonymous without the need for complex consensus mechanisms. Secure sum ensures that only aggregated updates are visible, preserving the privacy of individual agents' data. Both techniques are **highly efficient in terms of communication and computation, making them more suitable for environments where agility and scalability are key.** Thus, by opting for Gaussian noise and secure sum, the project achieves the same goals of secure and private learning while minimising network overhead, ensuring greater scalability, and simplifying the deployment process.

Another objective was to "support most widely used deep learning frameworks like TensorFlow, PyTorch or Caffe", which has been largely achieved. Currently, the tool supports TensorFlow/Keras, PyTorch, and Scikit-Learn, but not Caffe. This decision is based on both practical considerations and prevailing trends within the deep learning community.

Firstly, TensorFlow/Keras and PyTorch are by far the most widely adopted deep learning frameworks, with extensive user bases, active development, and rich ecosystems of tools and libraries. These frameworks are well-supported in both academic research and industry applications, making them the de-facto standards for most machine learning solitions. They also offer a range of high-level APIs and flexibility for researchers and developers, making them highly adaptable to a variety of tasks. On the other hand, while Caffe was once popular, especially for convolutional neural networks (CNNs), its usage has significantly declined over time. In fact, no releases have been launched since 2017.[38] The latest trends in deep learning show a clear preference for TensorFlow/Keras and PyTorch,[39] largely due to their ease of use, flexibility, and ability to handle dynamic computation graphs. In contrast, Caffe's static graph architecture makes it less suitable for many modern applications. Additionally, community support for Caffe has waned, with fewer updates and contributions in recent years, meaning that it is no longer as actively maintained or developed as TensorFlow and PyTorch. By focusing on the most commonly used frameworks, we ensure that the system remains aligned with the tools preferred by the majority of deep learning practitioners. **This decision enhances the practicality and scalability of the framework, allowing users to work with familiar tools while maintaining flexibility and performance** in model development.

Regarding the claim that "state-of-the-art Natural Language Processing (NLP) libraries like Hugging Face would be integrated to process textual content from websites or social media channels," please refer to Deliverable 3.1, wherein it is explained that no textual data is to be processed within the SL infrastructure and as such, the use of SOTA NLP has, instead, been used within the Sentiment Analysis module, particularly the mentioned libraries.

4.3 Achievement of KPIs, KRs, and TOTs

The SL framework, as stated in the GA, is expected to "facilitate training ML models for predicting offline and online crime caused by D&FN, tailored to the specific needs of police authorities."[40] This requirement has been fully covered, as a crime forecasting model is included in the framework and has shown the usefulness of the tool to produce more powerful and robust ML models using data from different data sources. In addition, the output of the framework has been linked to the Dynamic Flows Modeler, an additional layer of forecasting which studies the spread of D&FN alongside crime. As stated in the GA, the framework was thought as "a completely decentralised approach", which "will guarantee compliance with existing regulations for data protection and minimise the attack surface", and "will facilitate the dynamic and agile collaboration between multiple LEAs throughout the European geography since the role of a central entity will be not needed." All

[38] Berkeley Vision and Learning Center, 'Releases of Caffe: A Fast Open Framework for Deep Learning', *GitHub repository*, n.d., https://github.com/BVLC/caffe/releases.

[39] 'Restack, 'PyTorch vs TensorFlow vs Keras vs Theano vs Caffe: Detailed Comparison,' *Restack.io*, n.d. https://www.restack.io/p/pytorch-answer-vs-tensorflow-vs-keras-vs-theano-vs-caffe

[40] 'Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021.

these statements have been covered, as the swarm methodology itself involves the deployment of decentralised agents, in the absence of a central server, in a way that the data produced in each of the agents never leave the specific agent.

Regarding the objective of SL for "increasing SOTA learning speed by at least 50%", we have carried out a comparative analysis. We have chosen Flower[41] as the benchmark framework due to its prominence and utilisation across diverse federated learning applications. Flower has been recognised as one of the top federated learning frameworks in comprehensive comparative studies,[42] scoring highly in several evaluation categories including usability and interoperability.[43] This makes Flower an excellent candidate for benchmarking against Fleviden framework, as it represents a widely adopted solution in the federated learning landscape.

The analysis involved carrying out an experiment running the same federated learning experiment on both the Flower and Fleviden frameworks using identical datasets, federated learning topologies (vanilla FL), number of training rounds (8), number of clients (4), hyperparameters like the number of training epochs in the clients (2) or the batch size (128), etc., to compare total execution times. The results indicated a training time of 443.79 seconds for Flower and 267.95 seconds for Fleviden. Consequently, Fleviden has demonstrated a learning speed that is approximately 65.62% faster than Flower. Talking about the R&I maturity of the framework, the requirement was to extend the Codex AI ATOS tool "with new capabilities targeting the specific needs of police authorities to detect disinformation and fake news". However, when the project started, swarm capabilities did not fit within that tool, so the development began at ground zero. Because of that, the expected TRL was reduced to 6, as the technology has been demonstrated in a relevant environment and properly integrated with a wide platform (i.e., FERMI).

Regarding the contribution to the expected impacts set out in the work programme destination, the GA states that "the SL module that [is] using a complete decentralized approach to train the ML module of FERMI framework would facilitate the dynamic and agile collaboration between multiple LEAs, and increase the predictive capabilities of offline and online crimes introduced by D&FN by more than 40%.". In this context, it is important to note that the SL infrastructure helps the Dynamic Flows Modeler in reaching the forecasts of offline-crime by providing part of the essential input data needed for its operation. This support is central to the recent improvements, which have shown a remarkable 40% enhancement in performance metrics. This underscores the importance of both the Swarm Learning infrastructure and the Dynamic Flows Modeler in achieving this advancement.

Regarding the additional development targets (i.e., the TOTs), the RMSE and MAE were chosen for the crime predictor in the Swarm Learning module because they are standard metrics for assessing ML models in regression tasks and offer valuable insights into model performance. The RMSE, targeted to be under 50 crimes, provides an understanding of how the model manages the overall prediction set and is sensitive to outliers, while the MAE, aimed to be under 40 crimes, offers a more straightforward and intuitive interpretation, allowing for a direct assessment of how much the model's predictions typically deviate from the actual values.

With the current model (see Subsection 4.2.2), the metrics obtained show an RMSE of 35.96 crimes and a MAE of 21.65 crimes for the best round of execution (Figure 12). These results are calculated using a separate dataset for validation in the agent which acts as server in each round. It is important to note that these results encompass the prediction of 11 different types of crimes, not just a single type of crime.

[41] Beutel, D.J., et al., 'Flower: A Friendly Federated Learning Research Framework', *arXiv preprint arXiv:2007.14390*, 2020. https://arxiv.org/abs/2007.14390.

[42] Riedel, P., Schick, L., von Schwerin, R., et al.,'Comparative analysis of open-source federated learning frameworks - a literature-based survey and review'. *Int. J. Mach. Learn. & Cyber*, 2024. https://doi.org/10.1007/s13042-024-02234-z

[43] Flower.ai, 'Comparison of Federated Learning Frameworks', *Flower.ai Blog*, n.d., https://flower.ai/blog/2024-07-22-fl-frameworks-comparison/

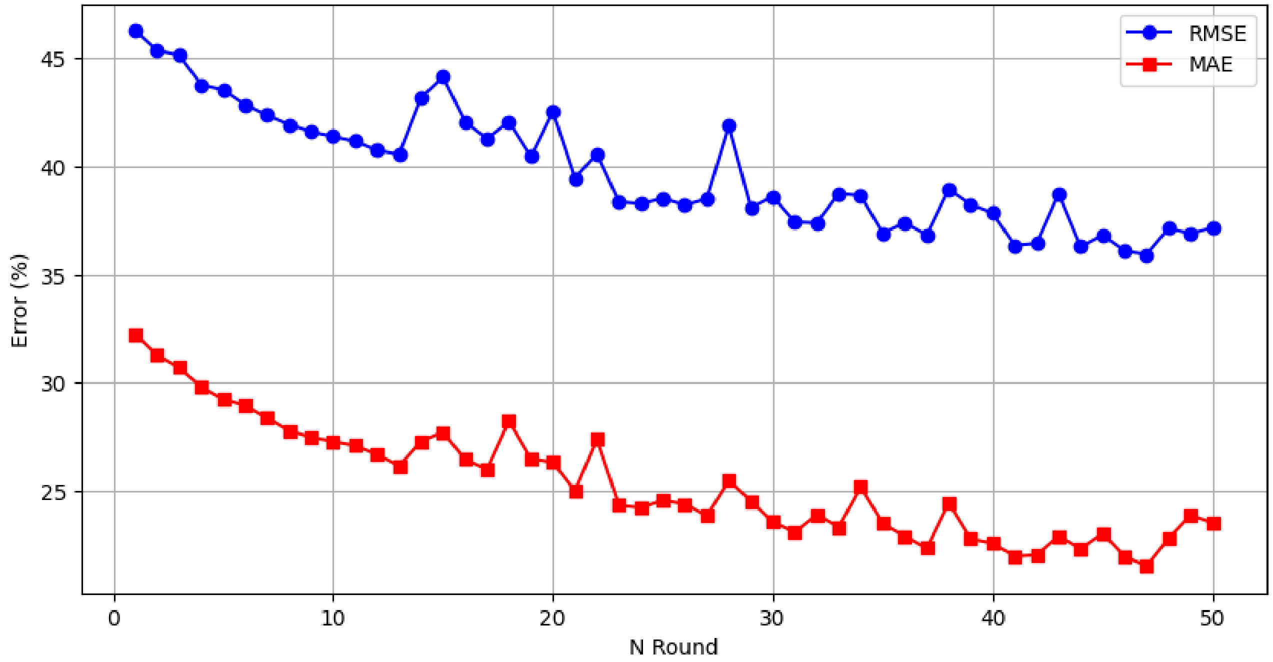


Figure 12: Results Obtained during an Execution of a Crime Forecast within the SL module

These figures represent approximate outcomes for a specific execution, as the performance of an AI model is not deterministic and can vary slightly from one execution to another. However, they tend to be within these ranges. As can be observed from the accompanying graph, the error metrics decrease as the model undergoes further training through successive rounds of swarm learning. **This trend highlights the effectiveness of the model's learning process over time, gradually improving its accuracy in crime prediction** as it adapts to the data provided during each round of swarm learning.

4.4 Versatility to Changing End-User Needs

The SL infrastructure we developed is highly flexible, allowing it to be adapted to meet a wide variety of end-user needs. This adaptability means that agencies from different regions or sectors can leverage the framework to address diverse machine learning challenges, as long as the data shared by all participating agents is of a similar nature and aimed at solving a common problem. This approach ensures that the technology is not restricted to a specific use case but can be tailored to fit new scenarios, end-users, and data sources.

Agencies interested in collaboratively training a unified, up-to-date ML model alongside LEAs from different regions can easily adapt the framework. By regularly incorporating the latest data and updating the model, these agencies can maintain the accuracy and relevance of their predictive tools, adjusting to new requirements and challenges as they arise. This ensures that the platform remains versatile and effective, regardless of variations in specific end-user needs.

The SL infrastructure provides LEAs with a powerful tool to address a wide array of challenges using their own data. This versatile platform allows agencies to collaboratively develop and train models tailored to specific problems they face, whether it involves crime forecasting, fraud detection, or any other unique case where data from different locations is available. By utilising standardised data—such as transaction histories or communications metadata – **LEAs can harness this technology to build predictive tools that enhance their operational effectiveness**. For instance, the framework can be applied to optimise responses to public safety alerts, manage large-scale event security, or optimise traffic incident response, provided the data used, like images or CSV files, is consistent in nature. This adaptability underscores the platform's capacity to meet diverse law enforcement needs while maintaining strict standards of privacy, data protection, and security.

To conclude with this section, it is important to highlight how ongoing user feedback has driven enhancements to the FERMI platform. These improvements include the integration of secure sum protocols, which enhance privacy, data protection, and trust by allowing the merging of AI model outputs while keeping individual contributions confidential. Furthermore, we have upgraded the user interface to more clearly illustrate the connections between digital and offline crime, facilitating the way law enforcement agencies interpret and utilise the data. To visually demonstrate these advancements, a video demonstration has been developed, showcasing the functionalities and underscoring the platform's capability to support secure and effective collaboration without the need for external data sharing.

4.5 SL Framework Summary

The SL framework is designed to offer a scalable software architecture that enables machine learning models to be trained close to where the data is generated. This approach ensures that the data remains secure and protected, as it is not transferred from its original location. By keeping data at the source, the architecture supports privacy compliance and addresses concerns associated with sharing sensitive information. Integrated within the FERMI platform, this technology works in combination with the Dynamic Flows Modeler (T3.1). Together, they empower the platform to analyse historical crime data from various independent LEA partners. With this setup, analyses about crime trends can be performed without compromising the privacy of the involved parties, as each LEA's data remains securely on their own servers, both **ensuring that sensitive information is not shared or centralised, and allowing agencies to work together, effectively**, without exposing their data to unnecessary risks or violating privacy regulations.

Section 4 outlined the updates and improvements introduced to the SL component in this second phase of development. The practical overview covered the various changes made, including updates to the framework's pods, adjustments to the AI model, the addition of a new inference service to facilitate integration with the Dynamic Flows Modeler, modifications to the statistical features used by the model, and a summary of the core functionalities promised and covered by the Swarm Learning tool.

Following this overview, we provided a technical explanation detailing how these enhancements were implemented, ensuring the tool aligns with specific performance goals. We also examined how the tool successfully meets various KPIs, KRs, and TOTs. Finally, we addressed the tool's adaptability to evolving end-user requirements, demonstrating its flexibility and potential for future enhancements within the swarm learning ecosystem. This comprehensive development ensures the tool remains robust, efficient, and ready to meet the demands of real-world applications.

5 Task 3.6 – The Sentiment Analysis Module

The Sentiment Analysis module analyses social media posts containing or considered D&FN, with the aim of providing end-users a perception of the emotional tone in said posts' content. The Sentiment Analysis module, in accordance with the GA, exploits bidirectional encoder representations from transformers (BERT) while ensuring the anonymisation of the posts, deletion of links, and processing of emoji characters to extract additional information. In doing so, "the classification [of] results of one specific instance are affected by both past and future instances,"[44] providing end-users a wholistic understanding of the content's sentiment.

Subsection 5.1 is structured into three main sections to provide a clear and detailed understanding of the work accomplished and its broader implications. Section 5.2 then analyses the technical specifications of the Sentiment Analysis module. It provides a detailed description of the model's training process, including the datasets employed and the methodological advancements made during the inference phase. This technical description demonstrates the robustness of the module and its alignment with project goals. Just as well, in this same subsection, the challenges encountered during development are addressed, outlining the strategic measures taken to overcome these obstacles and ensure alignment with the project's KRs, KPIs and TOTs (addressed in the proceeding subsection 5.3). Subsection 5.4 highlights the module's adaptability to shifting end-user requirements. It describes the modifications and enhancements made in response to feedback obtained during pilot testing. By showcasing these adjustments, this section underscores the tool's flexibility and its capacity to evolve in alignment with diverse operational needs, ensuring it remains relevant and effective in practical applications. Lastly, subsection 5.5 presents a concluding summary of the Sentiment Analysis module.

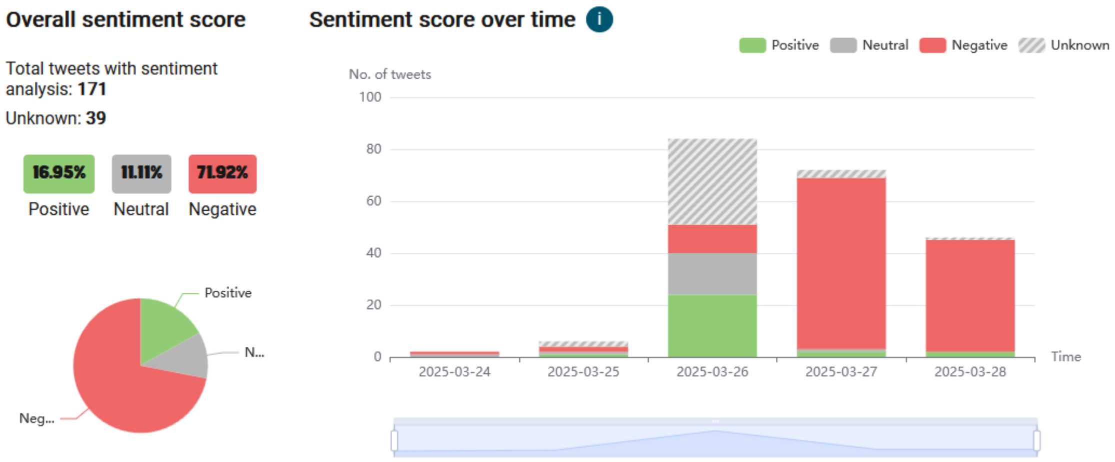
5.1 Practical Description

The Sentiment Analysis module is designed to assess and communicate to FERMI end-users the emotional disposition of social media posts regarding D&FN campaigns, thereby facilitating the identification of potential linkages between the spread of D&FN, in the online realm, and the risk of offline escalation and criminal behaviour. Its technological foundation is comprised of SOTA NLP and ML. **By scrutinising sentiment patterns embedded in social media content, the module unveils linguistic tones employed** by individuals involved in the dissemination of D&FN and, consequently, constitutes a significant step in the effort to assess the likelihood of D&FN-enabled offline actions, especially criminal activities.

The module utilises the power of the cutting-edge BERT language model, which is in line with the GA's requirement to "exploit the BERT model […] with a wide variety of NLP tasks,"[45] to dive into the vast realm of social media data (e.g., X posts' data graphs with highly influential nodes spreading disinformation)." The BERT model leverages context from both past and future words to make an estimate for a certain task. As further stated in the GA, experimenting with a bidirectional LSTM as a feature extractor was carried out.[46] However, based on the initial results of these experiments, this approach was discarded, and all progress in this deliverable has been achieved through the fine-tuning of BERT method. Additionally, the module offers sentiment analysis predictions for posts in German, Spanish, Finnish, Swedish and French utilising translator models.

The model takes as input the text of a post and outputs the predicted class of the post (0 for negative, 1 for neutral and 2 for positive), the predicted label of the post (negative, neutral and positive) and the certainty of the model for the prediction made. The overall predictions of the model across all nodes of the graph provided by the Spread Analyser are depicted under a specific section in FERMI's platform. Figure 13 shows the results of the sentiment analysis in a cumulative way. The pie chart shows the total results of the sentiment analysis making it easy to see which sentiment is the most dominant. The bar chart graph shows the results, and the number of posts involved the investigation per day, providing an overview of the sentiment over the progress of time

[44] 'Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021.
[45] Ibid.
[46] Ibid.



Overall sentiment score

Total tweets with sentiment analysis: **171**

Unknown: **39**

16.95% Positive | 11.11% Neutral | 71.92% Negative

Sentiment score over time

Figure 13: Example Results for Sentiment Analysis Module

5.1.1 Pilot Feedback and D3.1's Outstanding Steps

The Sentiment Analysis Module was tested by pilot-users in FERMI's second pilot, with very encouraging feedback being received. All the applicable pilot-user KPIs were met, with the only exception being UR029, which refers to the user's ability to evaluate the impact of a given D&FN campaign on public opinion. The provided feedback specifically noted that the X posts investigated during the pilot were not representative of public opinion. The limitations of a single platform in capturing a public opinion is a relevant concern and was rightly raised. As articulated in Deliverable 3.1, expanding the platform's capabilities beyond X was a part of the next steps to be taken, moving from the first versions of the technology offerings, which were tested using X, to the end-product versions, applicable to social media platforms beyond X.

Just as well, Deliverable 3.1 outlined that the Sentiment Analysis module, at the time of its writing, remained to be enhanced to achieve an accuracy performance of >90%. The objective of >90% is required by the GA.[47] To address this, additional resources were allocated to advancing the tool's capability and suitability to FERMI's use-cases. Moreover, the Sentiment Analysis module had yet to achieve TRL-6 as is to be demonstrated and evaluated for functionality and reliability in the operational environment of LEAs, which is the technical readiness benchmark stipulated by the GA. The challenges and limitations identified at the time of the 1st version of the module (domain general data, specificity versus accuracy), largely influenced the actions taken to enhance the model.

Indeed, domain-specific datasets were sourced to train models better suited to FERMI's use cases. The module now provides three models: **(1)** a general model, (the 1st version as reported in Deliverable 3.1), **(2)** a model focused on health-related topics, and **(3)** another centred on political social media posts. These models demonstrate improved performance metrics by addressing domain-specific limitations. Furthermore, the taken actions align with the first review's recommendations, to avoid overly specific modules, as use cases may evolve in the future.[48] Additionally, for the use-cases concerning left- and right-wing extremism, the general topic of political extremism has been selected to balance the trade-off of specificity versus accuracy, in line with the developmental choices of T3.1. As reported in subsection 2.2, a single model is used to address left- and right-wing extremism, for T3.1, as repeated trials during the development stage showed an insignificant variation between the results of the models trained on extremist D&FN, separated ideologically.

[47] Ibid.

[48] 'General Project Review Consolidated Report (HE) - Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01', *European Research Executive Agency*, 2024

Furthermore, the decision to explore integrating a 2-class model into the Sentiment Analysis module, as described in Deliverable 3.1, has been discontinued under the recognition of the importance of retaining the neutral class. A **special focus has been placed on making the Sentiment Analysis module more adaptable to different social media platforms** by including various types of text in our training datasets, such as headlines and comments. Importantly, the module's models have been evaluated on content from platforms other than X, confirming that the module is not platform-specific.

Lastly, having identified the challenge of multilingual posts, a translation workflow has been established to handle posts in languages other than English. The Sentiment Analysis tool provides support for several languages, such as German, Spanish, Finnish, Swedish and French. A literature review has also been conducted to propose more sophisticated multilingual approaches for future improvements.[49]

5.1.1.1 Results of the Second Round of Pilots

5.2 In the second round of piloting, the Sentiment Analysis Module was tested during pilot 2, wherein 7 active-duty LEA personnel, 5 non-active duty LEA staff, 11 LEA advisers, and 2 acquisition experts partook as pilot-users and all 23 completed evaluation questionnaires. Despite insufficient pilot-user satisfaction in the first round of pilots, for UR029, the second round saw a satisfaction rate of 95.65%, more than 30% above the target value. This marked improved is likely due to the introduction of explanatory texts in the user-interface, allowing end-users, in this case the pilot-users, to better understand the technology and how its results should be interpreted. A full description of the results of the second round of pilots can be found in Deliverable 5.3. Technical Description

The primary objective of the 2nd version of the Sentiment Analysis module is to offer a model with improved performance metrics in comparison to the 1st version, that is, a module that more accurately predicts the polarity of a given social media post. Its implementation follows the same methodological steps as presented in Deliverable 3.1, however, with greater emphasis on the training phase, as this procedure resulted in the improved model. The training is performed using again annotated datasets, with the difference that these datasets are domain-specific, rather than exclusively comprised of X posts. The improved models are then deployed for inference, in which case they provide sentiment analysis labels on new posts.

5.2.1 Training Phase

The training pipeline we use is the same as in the 1st version of the module with the main difference being the datasets we use for training. Figure4 illustrates the training workflow of the improved model. In this version, the same approach has been followed, that is, fine-tuning a RoBERTta model pretrained on the **3 class Stanford Sentiment Treebank (SST3)**,[50] as the feature-extraction approach seemed to have poorer performance.

[49] Ibid.

[50] 'RogerKam/roberta_fine_tuned_sentiment_sst3,' n.d. https://huggingface.co/RogerKam/roberta_fine_tuned_sentiment_sst3.

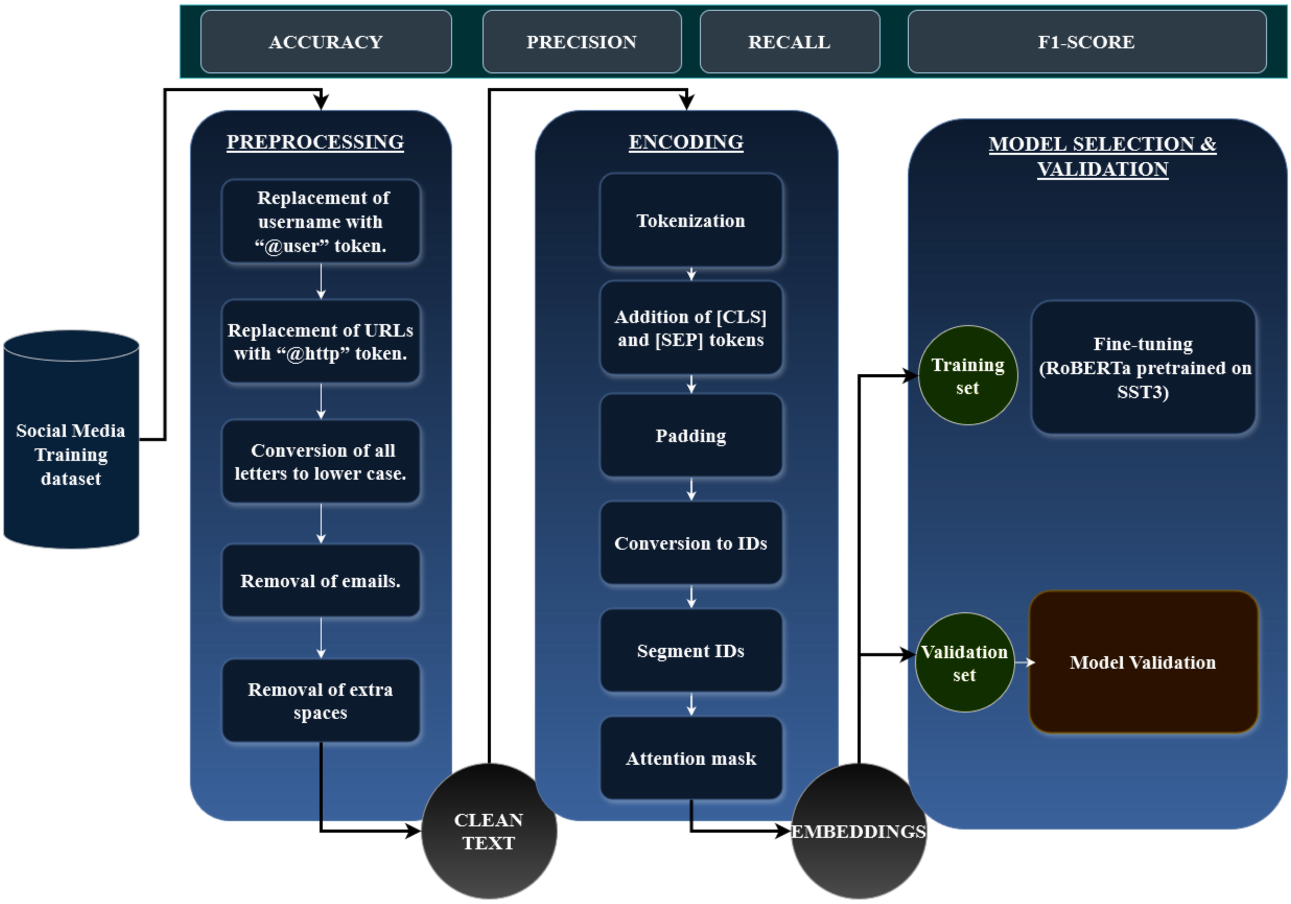

ACCURACY
PRECISION
RECALL
F1-SCORE
Social Media Training dataset
PREPROCESSING
Replacement of username with "@user" token.
Replacement of URLs with "@http" token.
Conversion of all letters to lower case.
Removal of emails.
Removal of extra spaces
CLEAN TEXT
ENCODING
Tokenization
Addition of [CLS] and [SEP] tokens
Padding
Conversion to IDs
Segment IDs
Attention mask
EMBEDDINGS
MODEL SELECTION & VALIDATION
Training set
Fine-tuning (RoBERTa pretrained on SST3)
Validation set
Model Validation

Figure 14: Step-by-step Overview of the 2nd of the Sentiment Analysis module's Training Phase

5.2.2 Training Data

Extensive research was carried out to identify several suitable data sources for the creation of the training datasets. The public health-related and politics datasets were formed by combining these sources, resulting in two unified datasets that provide sufficient records for the models to effectively learn from. This process is comprised of three main steps: data exploration and cleaning (identifying null values, duplicates, etc.), data processing (transforming the data into the same format), and identification of class distribution. Among the public health-related datasets were the following: COVIDSenti,[51] Healthcare Related Tweets,[52] Covid-19 Vaccine Tweets with Sentiment Annotation,[53] and sentiment analysis for Medical Drugs.[54]

5.2.2.1 COVIDSenti: a Large-Scale Benchmark Twitter Data Set for COVID-19 Sentiment Analysis

COVIDSenti is a large dataset, including 90.000 tweets about COVID, annotated as negative, neutral and positive. This dataset contains no null values, and the percentage of duplicates is very low (~0.2%). To create a unified format, it was mapped the original labels to 0 (negative), 1 (neutral) and 2 (positive), in accordance

[51] Naseem, U., et al., 'COVIDSenti: A Large-Scale Benchmark Twitter Data Set for COVID-19 Sentiment Analysis,' *IEEE Transactions on Computational Social Systems*, 2021. https://github.com/usmaann/COVIDSenti

[52] 'Healthcare Related Tweets for Sentiment Analysis,' *Omdena*, n.d. https://datasets.omdena.com/dataset/healthcare-related-tweets-for-sentiment-analysis

[53] 'Covid-19 Vaccine Tweets with Sentiment Annotation,' n.d. https://www.kaggle.com/datasets/datasciencetool/covid19-vaccine-tweets-with-sentiment-annotation

[54] 'Sentiment Analysis for Medical Drugs,' n.d. https://www.kaggle.com/datasets/arbazkhan971/analyticvidhyadatasetsentiment/data.

with TweetEval[55]. The class distribution of this dataset is depicted in Figure 15. It can be observed that this is a highly imbalanced dataset, with the neutral class being over-represented – with neutral X posts totaling 67385, negative X posts totalling 16335 and positive X posts totalling 6280.

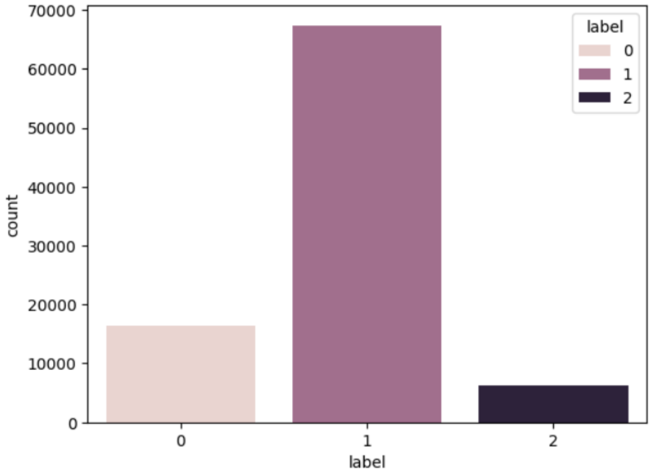


Figure 15: Class distribution of COVIDSenti

5.2.2.2 Healthcare Related Tweets for Sentiment Analysis

This dataset includes X posts related to public health in general, annotated with sentiment. It includes 23061 records with no duplicates or null values. To create a unified format, we have mapped the original labels to 0 (negative), 1 (neutral) and 2 (positive), in accordance with the TweetEval dataset.

The class distribution of this dataset is depicted in 11 – with 22,787 neutral, 68 egative and 146 positive X posts. Here as well, it can be observed that this dataset is highly imbalanced, with the positive and negative class being under-represented, especially the negative one. To tackle the imbalance in the 0 class, a small sample of negative X posts was generated using ChatGPT.[56] The input text we used as a starting point to generate these social media posts was: "Could you generate 30 similar X posts with negative sentiment label?" (providing some of the negative X posts of this dataset which can be found in **Fehler! Verweisquelle konnte nicht gefunden werden.**). Then ChatGPT generated another 30 records, hence 60 generated posts of the negative class were added to the dataset. The complete generated sample is shown in **Fehler! Verweisquelle konnte nicht gefunden werden.**. After the concatenation of the generated X posts with the original dataset, the final class distribution contains 22,787 neutral X posts, 128 negative X posts, and 146 positive X posts. The choice to generate only 60 samples is due to the current limitations in place when using ChatGPT.

[55] Rosenthal, S., et al., 'SemEval-2017 task 4: Sentiment analysis in Twitter,' *Proceedings of the 11th International Workshop on Semantic Evaluation*,' 2017.
[56] https://chatgpt.com/

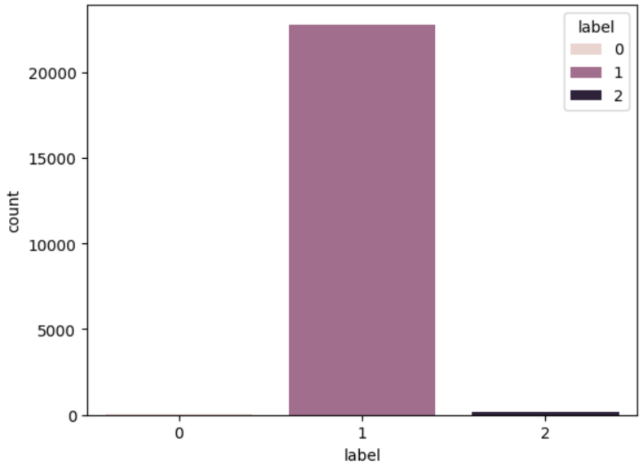


Figure 16: Class Distribution of Healthcare Related Tweets for Sentiment Analysis dataset

5.2.2.3 Covid-19 Vaccine Tweets with Sentiment Annotation

This dataset is a collection of X posts related to Covid-19 vaccines with human-annotated sentiments (negative, neutral, positive). The initial dataset included X posts about Pfizer/BioNTech, Sinopharm, Sinovac (both Chinese-produced vaccines), Moderna, Oxford/Astra-Zeneca, Covaxin, and Sputnik V vaccines. The dataset includes 6000 records, and it has no null values. The number of duplicates is very small (0.1%) so the decision was made to leave them as is. To create a unified format, the original labels have been mapped to 0 (negative), 1 (neutral) and 2 (positive), again in accordance with TweetEval. The class distribution of the dataset is depicted in Figure 16. This is also a dataset with highly imbalanced classes, with 3,680 neutrel, 420 negative, and 1,900 positive posts.

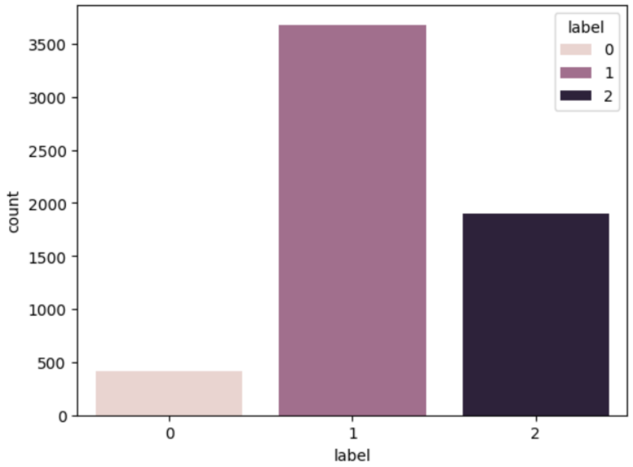


Figure 17: Class Distribution of Covid-19 Vaccine Tweets with Sentiment Annotation dataset

5.2.2.4 Sentiment Analysis for Medical Drugs

This dataset includes comments about patients where the topic is medications. The dataset includes 5279 records, with no null values and a small percentage of duplicate X posts (~2%). To create a unified format, the original labels have been mapped to 0 (negative), 1 (neutral) and 2 (positive), in accordance with TweetEval. The class distribution of the dataset is depicted in Figure 18 and is once again imbalanced, though in this case, the bias is towards the positive classification. The number of neutral X posts is 617, the X posts annotated as negative are 837 and the positive X posts are 3,825.

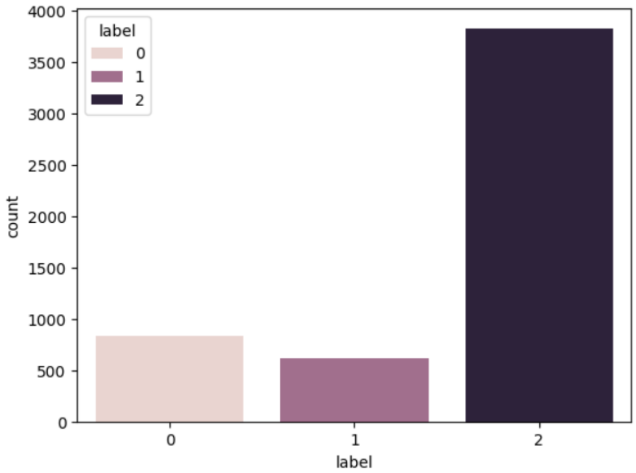


Figure 18: Class Distribution of Sentiment Analysis for Medical Drugs Dataset

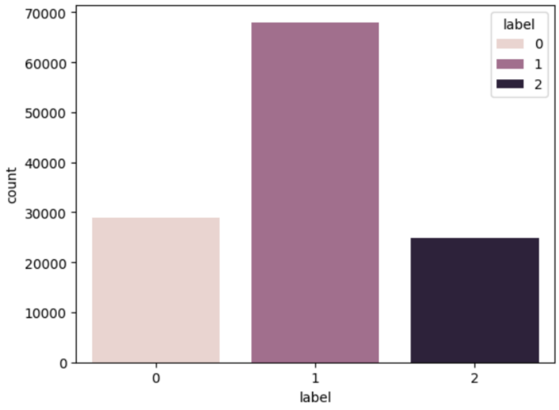
5.2.3 Public Health-Related Dataset's Split and Concatenation

The proceeding step involved each dataset being split using random sampling into train (70% of original data), validation (10% of original data) and test (20% of original data) data. Then these splits have been concatenated into a unified train, validation and test set, so for the training process three main datasets have been utilised containing health-related text (X posts and comments). The dimensions of the final unified dataset for health-related topics are shown in Table 8.

Table 8: Dimensions of the unified dataset for health-related topics.

Dataset	Records
train	121.822
validation	9.947
test	24.868

It should be noted that for the "Healthcare Related Tweets for Sentiment Analysis" dataset, the random sampling resulted in the absence of the negative class in some of the splits, so oversampling was used for this particular dataset in order to have a significant number of instances of each class in the training set. The class distributions of the training, validation and testing datasets are depicted in figures 19, 20, and 21, respectively. The class imbalance is still present in all dataset splits; however, this issue is tackled during training as in the 1st version of the module.



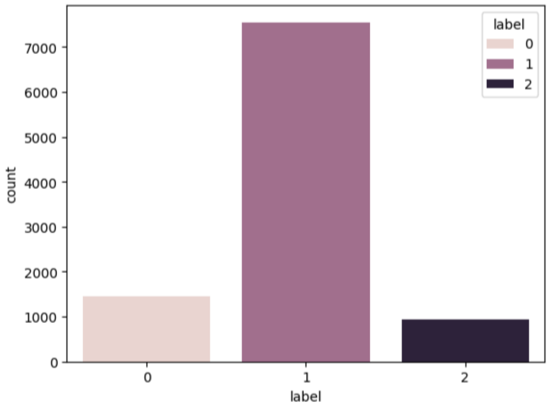


Figure 19 (Left): Class Distribution of Public Health-Related Training Set

Figure 20 (Right): Class Distribution of Public Health-Related Validation Set

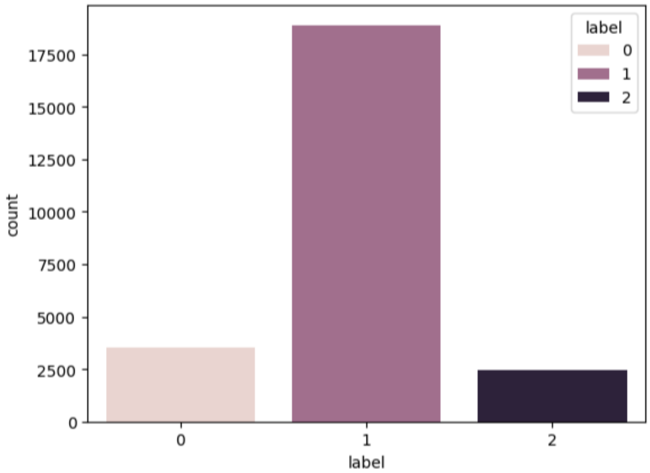


Figure 21: Class Distribution of Public Health-Related Test Set

5.2.4 Politics-Related Datasets

5.2.4 provides information about the datasets identified for the politics-related analysis. Each dataset is presented separately including its source and the results of our initial exploratory analysis and the process followed to create the unified dataset which was utilised for training the politics-specific model.

Extensive research was conducted to identify suitable datasets, tailored to the use cases of left and right extremism. However, datasets specifically targeting left and right-wing extremism, particularly in Europe, are scarce, especially in the context of sentiment analysis. Additionally, there was the requirement to develop a versatile and flexible tool, not too specific to the current use cases. Hence, it was decided to search for a broader domain, such as politics-related datasets that capture reactions to political debates, campaigns, and ideologies. Moreover, since political sentiment often shares underlying linguistic patterns regardless of the geographic context, datasets from other countries or continents were used. These datasets offer high-quality annotations, sufficient data volume, and nuanced expressions of political opinions, making them valuable for training. Their focus on divisive or ideologically charged discussions provides a relevant foundation for refining sentiment models in politically polarised contexts, compensating for the absence of directly applicable datasets. In particular, this applies to the Twittersphere in the context of the Republican Presidential debates amidst the rise of the alt-right.[57] Specifically, the following datasets were employed: First GOP Debate Twitter Sentiment,[58] SEN,[59] Politics-and-virality-twitter,[60] Political Sentiment Analysis,[61] and ECE143-Political-Sentiment-Analysis.[62]

5.2.4.1 First GOP Debate Twitter Sentiment

This dataset was originally derived using the Crowdflower's Data for Everyone library. It contains tens of thousands of X posts about the Republican Party's (also referred to as the Grand Old Party or GOP) debate in Ohio in early August 2016 and contributors were asked to do both sentiment analysis and data categorisation. Contributors were asked if the tweet was relevant, which candidate was mentioned, what subject was mentioned, and then what the sentiment was for a given tweet.

[57] Hawley, G., *Making Sense of the Alt-Right*, 2017, pp. 115-138. https://doi.org/10.7312/hawl18512-007.

[58] 'First GOP Debate Twitter Sentiment,' n.d. https://www.kaggle.com/datasets/crowdflower/first-gop-debate-twitter-sentiment?select=Sentiment.csv

[59] Katarzyna B., et al., 'A Dataset for Sentiment Analysis of Entities in News Headlines,' Procedia Computer Science, 2021. https://doi.org/10.1016/j.procs.2021.09.136

[60] Antypas, D., et al., 'Negativity spreads faster: A large-scale multilingual twitter analysis on the role of sentiment in political communication,' *Online Social Networks and Media*, 2023. https://doi.org/10.1016/j.osnem.2023.100242

[61] 'Political Sentiment Analysis,' n.d. https://www.kaggle.com/datasets/subhajournal/political-sentiment-analysis

[62] 'ECE143-Political-Sentiment-Analysis,' n.d. https://github.com/akashboghani/ECE143-Political-Sentiment-Analysis/tree/master/data

The dataset contains 13,871 records, with no null values and with 3469 duplicated X posts which were removed. Finally, 10,402 records remain. To create a unified format, we have mapped the original labels to 0 (negative), 1 (neutral) and 2 (positive), in accordance with TweetEval. The class distribution is depicted in Figure 22, including 6,084 negative, 2,645 neutral and 1,673 positive records.

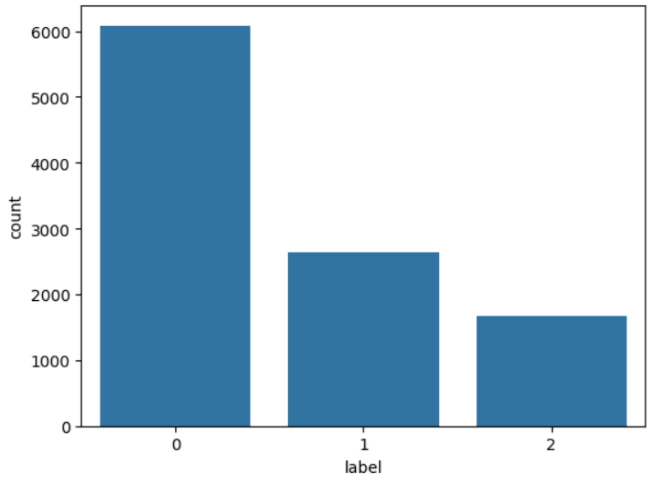


Figure 22: Class distribution of First GOP Debate Twitter Sentiment dataset

5.2.4.2 SEN - Sentiment analysis of Entities in News headlines

SEN is a novel publicly available human-labelled dataset for training and testing machine learning algorithms for the problem of entity-level sentiment analysis of political news headlines. This dataset consists of 3,819 human-labelled political news headlines coming from several major on-line media outlets in English. Each record contains a news headline, a named entity mentioned in the headline and a human-annotated label (one of “positive”, “neutral”, “negative”). Each headline-entity pair was annotated by a team of voluntary researchers or via the Amazon Mechanical Turk service.

For this dataset, a request to gain access to download the material was needed. After a minor process to concatenate the several offered datasets, the result was a set of 2,632 headlines of political news. The dataset contained 367 duplicates (~14%), which were removed, and 2,265 records remained. No null values were identified. However, there were some records (48) with unknown sentiment (labelled as ‘unk’), which were also removed. To create a unified format, the original labels were mapped to 0 (negative), 1 (neutral) and 2 (positive), in accordance with TweetEval. The dataset includes 863 negative, 971 neutral and 382 positive records and its class distribution is depicted in Figure 23.

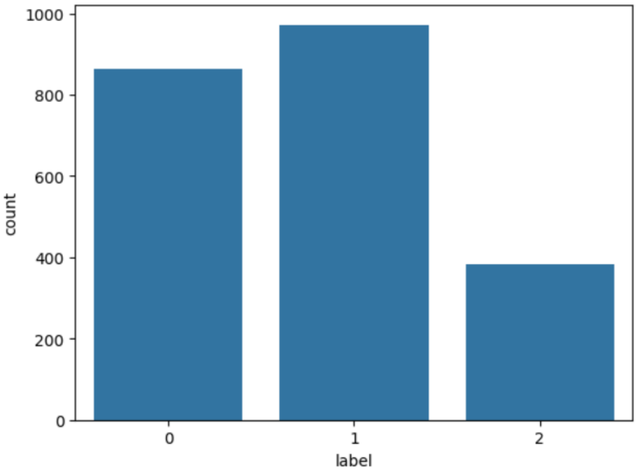


Figure 23: Class distribution of SEN - Sentiment analysis of Entities in News headlines dataset

5.2.4.3 Politics-and-virality-twitter

This dataset is the product of the paper “Negativity spreads faster: A large-scale multilingual twitter analysis on the role of sentiment in political communication.”[63] The dataset contains X posts in three languages (English, Spanish and Greek), from which we isolated the English dataset. The number of English records is 1000, from which 36 records have unknown labels, so we remove those. The dataset has no null values and only 1 duplicate which we remove. The remaining records are 963 rows. To create a unified format, we have mapped the original labels to 0 (negative), 1 (neutral) and 2 (positive), in accordance with TweetEval. The class distribution is depicted in Figure 24 and is comprised of 253 negative, 240 neutral and 471 positive records.

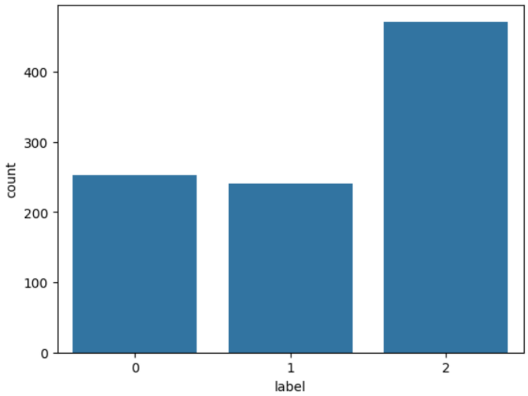


Figure 24: Class Distribution of Politics-and-virality-twitter dataset

5.2.4.4 Political Sentiment Analysis

This dataset has been collected by extracting hashtags from Twitter. The emphasis has been on the Russia-Ukraine war and the underlying sentiment of global X users who published X posts. Thus, in this context, the hashtags mentioned in Table 9, have been selected to fetch the data from the tool. To extract the sentiment label, TextBlob[64] (for the extraction of polarity and subjectivity) was used.

Table 9: Hashtags used for collecting the X posts of Political Sentiment Analysis dataset

Hashtags
#ukrainewar
#russianattack
#russian navy
#russianarmy
#prayforukraine
#NATO
#SaveUkraineNow
#ukraineunderattack
#ukrainecrisis
#StopPutinNOW
#ukraineconflict
#StopTheWar
#StopRussia

[63] Antypas, D., et al., ‘Negativity spreads faster: A large-scale multilingual twitter analysis on the role of sentiment in political communication,’ *Online Social Networks and Media*, 2023. https://doi.org/10.1016/j.osnem.2023.100242
[64] ‘TextBlob: Simplified Text Processing,’ n.d. https://textblob.readthedocs.io/en/dev/

The dataset contains 2,430 records, 965 of which are duplicates so we removed those. No null values were identified, so the remaining records are equal to 1465, of which 181 are negative, 978 are neutral, and 306 are positive. To create a unified format, in ensure coherence, the original labels were mapped to 0 (negative), 1 (neutral) and 2 (positive), as in TweetEval. The class distribution is depicted in Figure 25.

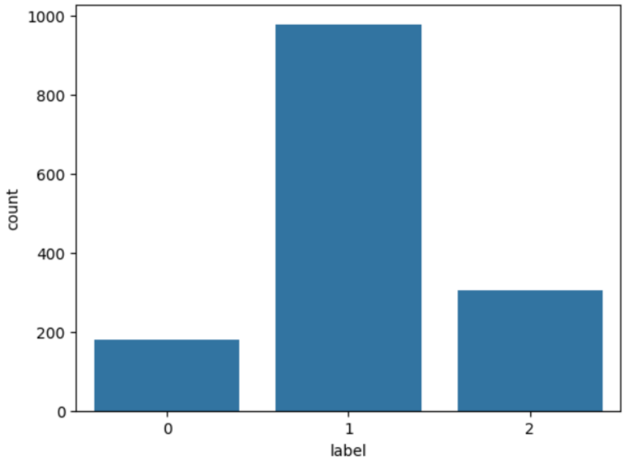
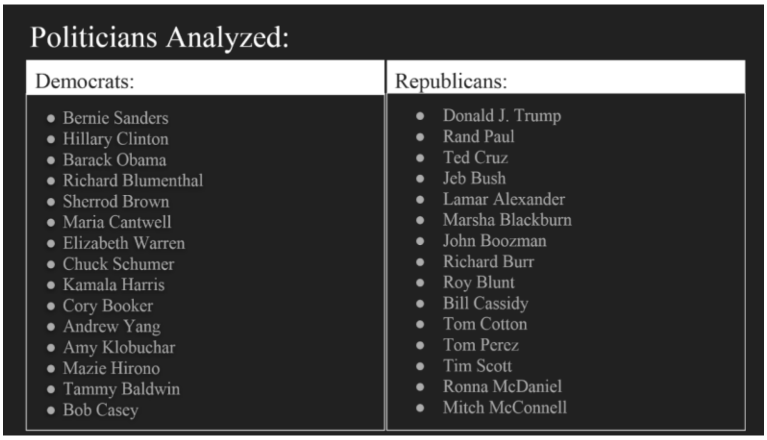


Figure 25: Class distribution of Political Sentiment Analysis dataset

5.2.4.5 ECE143-Political-Sentiment-Analysis

This dataset contains X posts from well-known Democrats and Republicans, the two primary parties in the United States. The list of names that are searched is depicted in Figure 26. The sentiment labels have been assigned using a Lexicon approach, scoring n-grams and finding polarity and subjectivity scores for a given string. Since the dataset contains the polarity score returned by TextBlob,[65] an additional rule was applied to derive the positive, neutral and negative labels for sentiment. Polarity lies between [-1,1], where -1 defines a negative sentiment and 1 defines a positive sentiment and in TextBlob negation words reverse the polarity.[66] Hence, the records with polarity score lower than or equal to -0.1 were assigned the negative label, the ones with polarity higher than or equal to 0.1 were assigned as positive and the records with polarity score in between were assigned the neutral class. These limits were fine-tuned after comparing TextBlob labels with human-annotations from TweetEval.



Politicians Analyzed:

Democrats:

- Bernie Sanders
- Hillary Clinton
- Barack Obama
- Richard Blumenthal
- Sherrod Brown
- Maria Cantwell
- Elizabeth Warren
- Chuck Schumer
- Kamala Harris
- Cory Booker
- Andrew Yang
- Amy Klobuchar
- Mazie Hirono
- Tammy Baldwin
- Bob Casey

Republicans:

- Donald J. Trump
- Rand Paul
- Ted Cruz
- Jeb Bush
- Lamar Alexander
- Marsha Blackburn
- John Boozman
- Richard Burr
- Roy Blunt
- Bill Cassidy
- Tom Cotton
- Tom Perez
- Tim Scott
- Ronna McDaniel
- Mitch McConnell

Figure 26: Political Actors Searched in X within ECE143-Political-Sentiment-Analysis

[65] 'TextBlob: Simplified Text Processing,' n.d. https://textblob.readthedocs.io/en/dev/.

[66] Ibid.

The dataset originally contains 95,748 X posts. There is a column indicating the language, so we filter for English X posts only. The records are reduced to 95,601. The dataset contains 3081 null values in Polarity column, which we drop. Additionally, there are 627 duplicate posts, which we also drop. The final number of X posts in the dataset is 91,893. To create a unified format, the original labels were mapped to 0 (negative), 1 (neutral) and 2 (positive), in accordance with TweetEval. The classes distribute accordingly, 16,236 negative, 20,919 neutral, and 54,738 positive records.

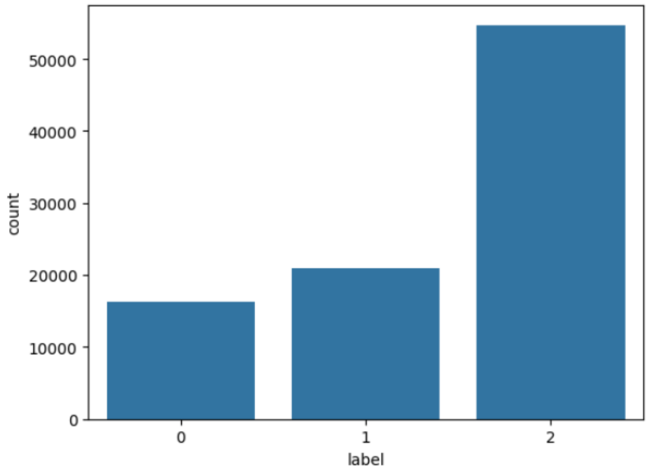


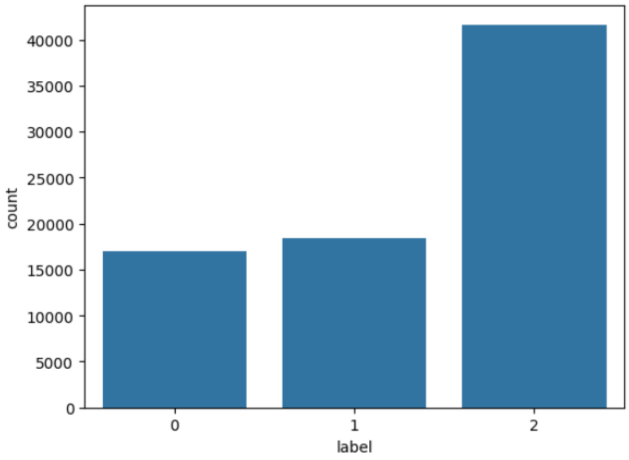
Figure 27: Class Distribution of ECE143-Political-Sentiment-Analysis Dataset

5.2.5 Politics-Related Datasets' Split and Concatenation

Proceedingly, each dataset was split using random sampling into train (70% of original data), validation (10% of original data) and test (20% of original data) data. Then, they were concatenated into a unified train, validation and test set, so for the training process they result in three main datasets containing politics-related text (X posts and headlines). The dimensions of the final unified dataset for politics-related topics are shown in Table 10. The class distributions of the training, validation and testing datasets are depicted in figures 28, 29, and 30, respectively. It is important to note that the class imbalance is still present in all dataset splits; however, this issue is tackled during training as in the 1st version of our module. In the training set, the class distribution is 16,985 negative, 18,379 neutral and 41,634 positive posts, while the validation set is comprised of 1,854 negative, 2,140 neutral and 4,560 positive posts. Lastly, the test set contains 4,777 negative, 5,235 neutral and 11,376 positive posts.

Table 10: Dimensions of the Unified Dataset for Politics-Related Topics

Dataset	Records
train	76,998
validation	8,554
test	21,388



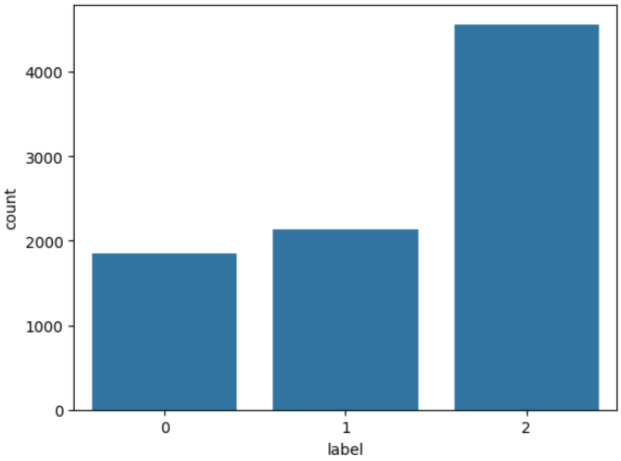


Figure 28 (Left): Class Distribution of the Training Set

Figure 29 (Right): Class Distribution of the Validation Set

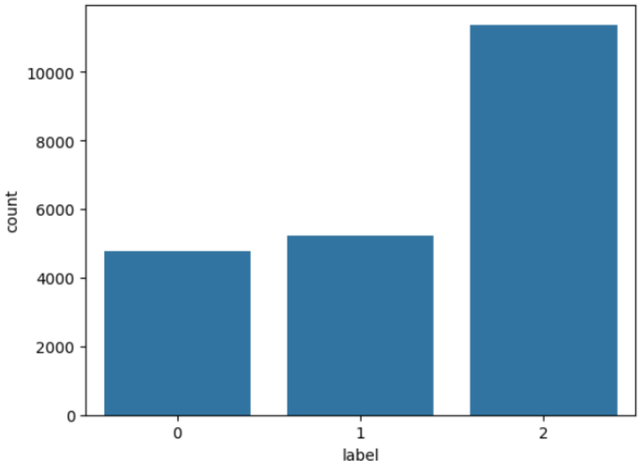


Figure 30: Class Distribution of the Test Set

5.2.6 Pre-Processing and Encoding of Data

As in the previous version of the Sentiment Analysis module's model, the first step of training is the data pre-processing and transformation to a format suitable to the model. The experiments conducted in the earlier iteration revealed that the moderate data cleaning resulted in the best metrics, which involves **(1)** replacement of username with "@user" token to eliminate any referenced users in the tweet and fully anonymise the text (in full compliance with data protection standards, see D7.1 and D7.2 for further information on this); **(2)** replacement of URLs with "@http" token to remove all URLs; **(3)** conversion of all letters to lower case; **(4)** removal of emails, and **(5)** removal of extra spaces**.**

Next, all texts are encoded in several steps. First, the text is tokenised, essentially text is split into smaller pieces called tokens. Then, classification and separator tokens are added to the text. To ensure consistency, inputs are padded to a fixed length. The tokens are then converted into their corresponding IDs. Segment IDs are assigned to each token, allowing for differentiation between different parts of the text. Following this, an attention mask is generated. This mask helps distinguish actual words (non-padding tokens) from padding tokens. Once all these steps are completed, the input data is formatted to ensure compatibility with the BERT model. Finally, multiple input examples are organised into batches. This batching process optimises efficiency during both the training and inference phases. The encoding process is performed by a tokeniser which is paired with a model. Moreover, as in the previous iteration, the encoding length remains 512 tokens maximum to ensure the inclusion of complete posts or sentences within our training dataset, thus preserving data integrity and model compatibility.

5.2.7 Model Selection and Evaluation

The preliminary work conducted during the first cycle of experiments revealed that the fine-tuning method produced models with higher performance. Consequently, in the current advancements we experimented only with the fine-tuning method and more specifically we utilised the best configuration in terms of performance metrics. This configuration includes a RoBERTa model pretrained on an SST3 dataset, applying a moderate data cleaning and preprocessing pipeline, along with a weighted loss function to address the class imbalance in our datasets. The hyperparameters that produced the best results in the first version are depicted in Table 11. These hyperparameters were also used in the training process of the current advancements.

Table 11: Hyperparameters for the (1st Version) Sentiment Analysis Module

Learning Rate	1.4305135307339992e-06
Weight-Decay	5.188348810329188e-05
Num-train-Epochs	31
Optimizer	adafactor
Per Device Train Batch Size	8
Per Device Eval Batch Size	12
Seed	42
Data-Seed	42
No Frozen Layer	0

5.2.8 Public Health-Related Model Results

The evaluation metrics of the health-related model are depicted in Table 12. We can observe that **the use of domain specific data has improved the model's performance significantly**, compared to the model trained and evaluated on TweetEval.

Table 12: Evaluation Metrics on the Public Health-Related Training and Validation Set

Eval Accuracy	Eval F1	Eval Loss	Eval Recall	Eval Precision	Train Accuracy	Train F1	Train Loss	Train Recall	Train Precision
0.908	0.847	0.429	0.870	0.827	0.944	0.939	0.376	0.945	0.933

Figures 31 and 32 depict the loss and accuracy during the training process. It is evident that the model is well generalised, as **the loss during training (blue line) and the loss during evaluation (orange line) do not exhibit large differences.** Similarly, the training curves are closely aligned, enhancing the evidence of the model's robustness.

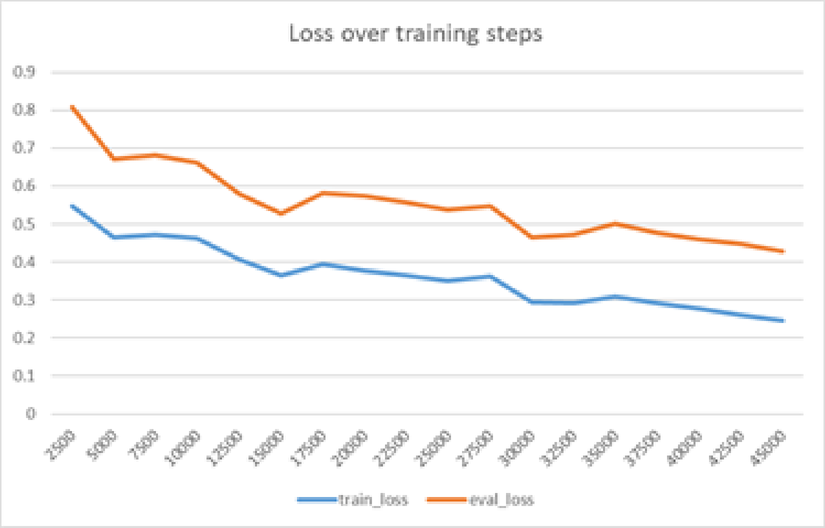


Figure 31: Loss of the Public Health-Related Model during the Training Process

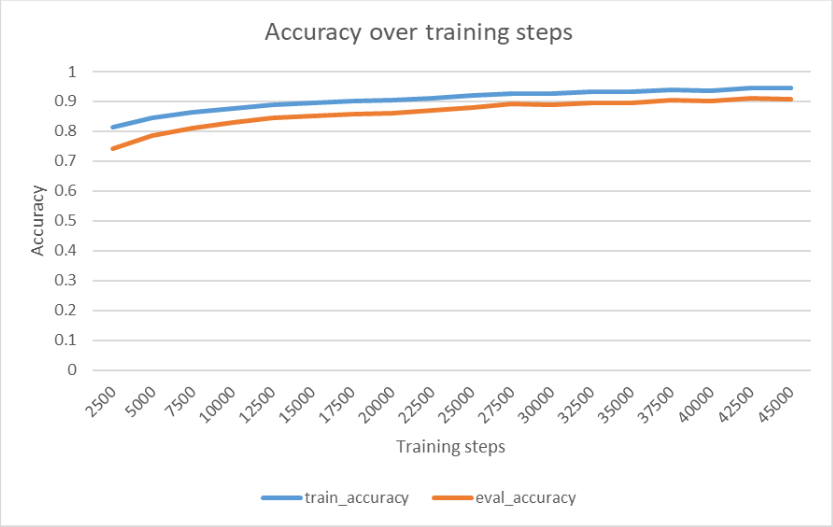


Figure 32: Accuracy of the Public Health-Related Model during the Training Process

To evaluate further our health-related model, the testing set previously built was used as completely unseen data. The metrics on this split are depicted in Table 13. As observed, the test metrics are aligned with the training and validation results.

Table 13: Evaluation Metrics on Public Health-Related Test Set

Test Accuracy	Test F1	Test Recall	Test Precision
0.897	0.832	0.862	0.808

Following the evaluation of the public health-related model, a final round of training was conducted using all available data. Specifically, we combined the training and test splits, resulting in a training dataset of 146,690 records, while maintaining the validation set at 9,947 records. This approach allows the public health-related model to learn from a larger dataset, which typically leads to a more robust and generalised model. This final version is the one deployed in the FERMI platform for sentiment analysis on posts related to health topics. The training metrics for this model are presented in Table 14, while the loss and accuracy during training steps are depicted in figures 33 and 34, respectively, with both indicating that the model is robust and has generalized well.

Table 14: Evaluation Metrics on the Public Health-Related Training and Validation Set - All Available Data

Eval Accuracy	Eval F1	Eval Loss	Eval Recall	Eval Precision	Train Accuracy	Train F1	Train Loss	Train Recall	Train Precision
0. 909	0.845	0. 457	0. 858	0. 834	0. 936	0. 927	0. 417	0. 930	0. 924

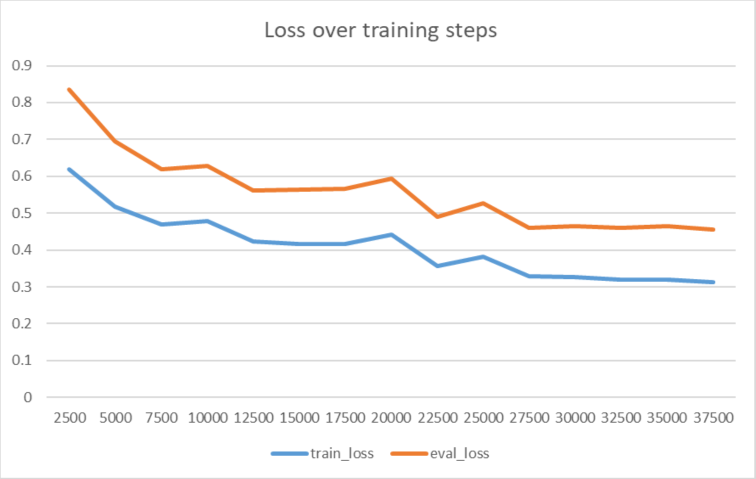


Figure 33: Loss of the Public Health-Related Model during the Training Process – All Available Data

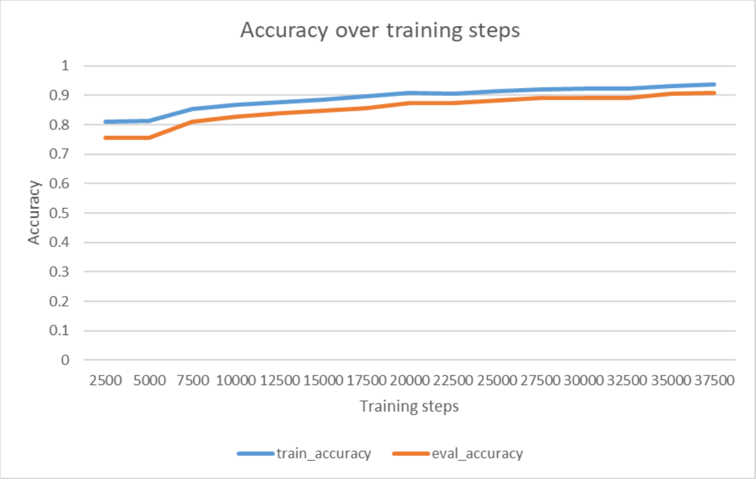


Figure 34: Accuracy of the Public Health-Related Model during the Training Process – All Available Data

5.2.9 Politics-Related Model Results

The model on the politics-related dataset was similarly trained and evaluated. The results of the training process are depicted in Table 15. It is observed that the politics-related model has also significantly improved metrics than the generalised model.

Table 15: Evaluation Metrics on the Politics-Related Training and Validation Set

Eval Accuracy	Eval F1	Eval Loss	Eval Recall	Eval Precision	Train Accuracy	Train F1	Train Loss	Train Recall	Train Precision
0. 844	0.816	0. 491	0. 819	0. 814	0. 878	0. 857	0. 600	0. 861	0. 853

Figure 35 and 36 depict the loss and accuracy of the politics-related model during the training process. It is evident that the model has not overfitted, since in both cases the training (blue line) and the evaluation (orange line) curves do not showcase large differences across training steps.

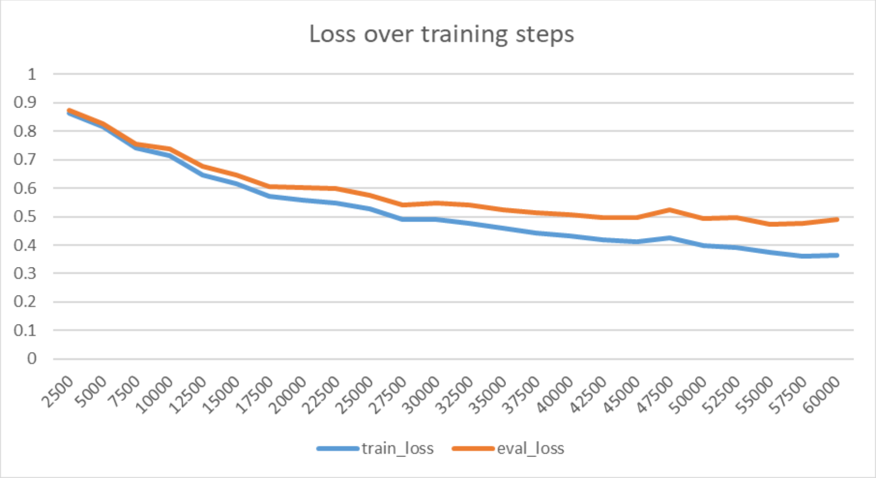


Figure 35: Loss of the Politics-Related Model during the Training Process

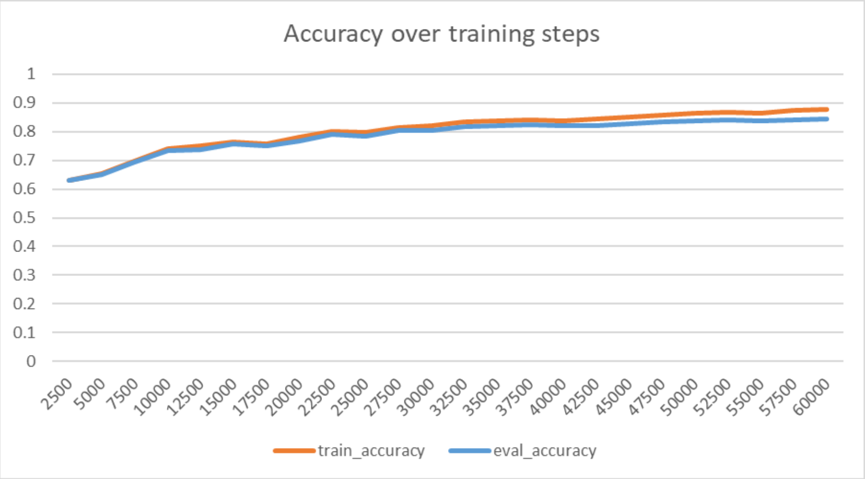


Figure 36: Accuracy of the Politics-Related Model during the Training Process

The metrics of the politics-related model on the test split created are presented below (Table 16), to evaluate its performance on unseen data. The model's performance on unseen data is aligned with the metrics achieved during training, which strengthens the confidence in the robustness of our model.

Table 16: Evaluation Metrics on Politics-Related Test Set

Test Accuracy	Test F1	Test Recall	Test Precision
0. 824	0. 798	0. 808	0. 835

Following the evaluation of the politics-related model, a final round of training was conducted using all available data. Specifically, the training and test splits were combined, resulting in a training dataset of 98,386 records, while maintaining the validation set at 8,554 records. **This approach allows the politics-related model to learn from a larger dataset, which typically leads to a more robust and generalised model**. This final version is the one deployed in the FERMI platform for sentiment analysis on posts related to political topics. The training metrics for this model are presented in Table 17. Additionally, the loss and accuracy during training steps are depicted in figures 37 and 38 respectively, both of which indicate that the model is robust and has generalised well.

Table 17: Evaluation Metrics on the Politics-Related Training and Validation Set - All Available Data

Eval Accuracy	Eval F1	Eval Loss	Eval Recall	Eval Precision	Train Accuracy	Train F1	Train Loss	Train Recall	Train Precision
0. 825	0.791	0. 541	0. 799	0. 797	0. 838	0. 806	0. 650	0. 814	0. 810

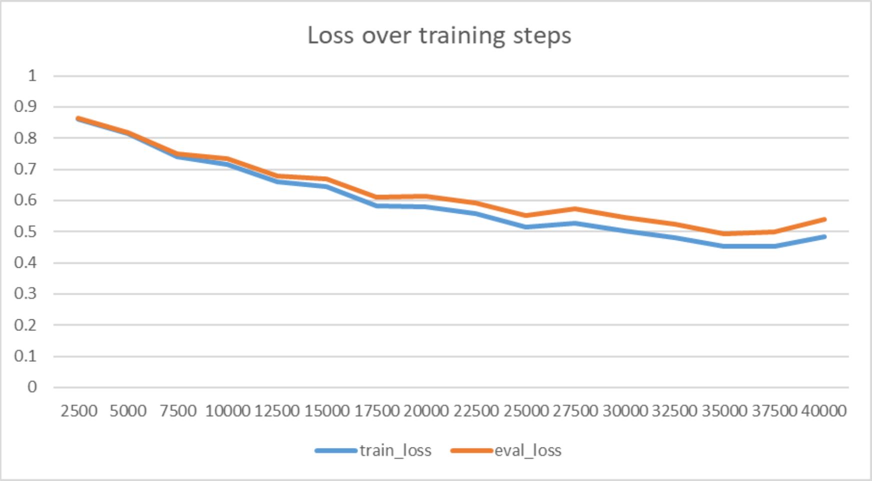


Figure 37: Loss of the Politics-Related Model during the Training Process - All Available Data

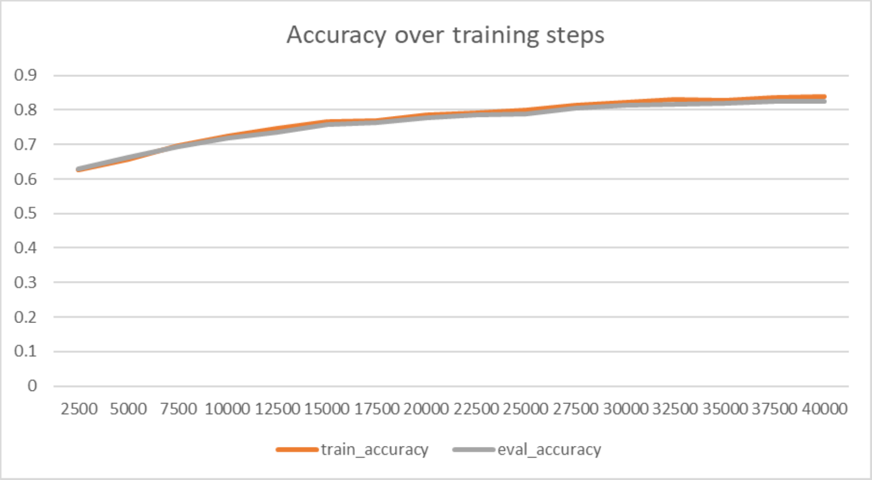


Figure 38: Accuracy of the Politics-Related Model during the Training Process - All Available Data

5.2.10 Interference Phase

As in the previous version of the Sentiment Analysis module (reported in Deliverable 3.1), the inference mode is comprised of pipelines for data ingestion and preparation, the serving of the trained models and the output generation. The process is very similar to the earlier iteration of the module, where the input is the Spread Analyser's generated graph. The module analyses the text of each node separately, pre-processes the post, performs encoding and finally the model predicts its sentiment. The structure of the input is maintained, but now each node contains also **the predicted sentiment label and the associated accuracy probability**. Moreover, the text of the post is removed before the return of the results due to legal limitations. The details of the implementation of the inference pipeline can be found in Deliverable 3.1.

The changes in this version include the addition of one extra field in the input derived from the Spread Analyser, the use case number. This additional field is used for loading the suitable model for predictions;

when the user specifies that the post under consideration is health-related, this information is passed to the Sentiment Analysis module and the health-related model is utilised. Similarly, the left- and right-wing extremism use cases are covered by the politics-related model. Additionally, a change in the translation workflow was implemented. In the previous version, the module would load the translator of the language of the user-provided post. This implementation introduces some limitations; when the graph includes posts in a language different than the country under consideration and English, then this post is not translated, and no sentiment is assigned. In this version, the module offers translation from five languages to English, according to the nationality of the project LEAs and Police Colleges.

The available languages are German, French, Swedish, Finish and Spanish. Nevertheless, we believe that this streamlined pipeline balances the trade-off between efficiency and effectiveness, which is crucial for maintaining the system's overall performance reliability while simultaneously providing users with as many actionable insights as possible. The workflow of the Sentiment Analysis module can be analysed in several steps. Firstly, the output graph of the Spread Analyser is loaded. At this point the graph has added information from the Spread Analyser, the bot detection field and the influence score, but no information concerning sentiment analysis. Then, the module processes each node separately, so for each node the text is extracted and translated if needed and if the language of the post is among the available ones (German, French, Swedish, Finish, Spanish.), pre-processed and encoded to be used by the Sentiment Analysis model. Next, the model makes a prediction, and three new fields are added to each node, the prediction_class, which is the numeric representation of the prediction (0 for negative, 1 for neutral and 2 for positive), the prediction_label, the textual representation of the label (negative, neutral and positive) and the prediction_probability, which is the certainty of the model for the prediction made. Lastly, the module returns the enriched model to the gateway for storage.

5.2.11 Challenges and Limitations Faced

5.2.11.1 Data Availability

Training a model is highly dependent on the dataset utilised in the process. Although datasets can be found for a wide variety of tasks and domains, their quality is not sufficient for training a model; they may include many null values or duplicates, have no labels for the task under consideration, or even have a considerable class imbalance. The number of high-quality datasets decreases even further when one considers that they also need to be specific to a particular domain. In our case, extensive research was conducted to identify datasets containing social media posts, maximising quality and balancing the need for quality with acquiring a sufficient quantity for training.

To address the challenge of unlabelled datasets, there are several alternatives such as human annotation (which is very extremely time-consuming and resource-intensive),automatic annotation tools (such as lexicon-based approaches (e.g., TextBlob), and pretrained machine learning models. While models, including those created during this project, can be used for annotation, they come with their own limitations. Models are often trained on domain-specific datasets and can be prone to bias or overfitting. They may not generalise well across different domains, potentially leading to inaccurate or inconsistent annotations. In contrast, TextBlob uses a lexicon-based approach that applies predefined rules and word sentiment scores. As such, it provides a more neutral baseline for annotation and, since it is not trained on specific datasets, is less susceptible to biases inherent in machine learning models.

In this context, datasets that have been annotated with TextBlob have been utilised. Despite its simpler mechanism, it offers certain advantages, particularly when the risk of model-induced bias needs to be minimised. To evaluate the difference of TextBlob from human annotation, TweetEval was utilised. The process included running Textblob on TweetEval and comparing the resulting labels to the existing human annotated labels of the dataset. The results revealed that approximately 52% of the TextBlob labels aligned with human annotations. Although this is not perfect, it demonstrates that TextBlob provides a reasonable baseline, especially when human annotation is not feasible. Furthermore, using data annotated with TextBlob allows to complement human-annotated data without introducing the same model's bias back into the training set, which could compromise the overall model's performance. While the ideal scenario involves full human annotation, the inclusion of TextBlob-annotated data helps ensure a larger, more diverse dataset, which is critical for reaching performance metrics such as >90% accuracy.

5.2.11.2 Social Media Platform Agnostic

In the 1st version of the module the challenge posed by different social media platforms has been analysed, as each one has its special characteristics (such as different character limits) resulting in users developing unique linguistic characteristics according to the platform. Consequently, having a generalised model across all platforms, could potentially lead to lower performance metrics.

The Sentiment Analysis module is generic by definition, meaning that BERT-based models can analyse any given text independently of the source. However, certain limitations are present, such as the difference between the maximum length of text used during training and the maximum length of the given text for prediction, which would have an impact on the model's performance.

To evaluate the model's possible performance decrease when the training was conducted using posts from one platform and predictions are made on posts of other platforms, two datasets were used, **Twitter and Reddit Sentimental analysis Dataset**[67] and **Social Media Sentiments Analysis Dataset,[68]** which contain posts from several platforms. In the case of FERMI, the vast number of posts come from X, so a model trained only on tweets will be evaluated and used to predict posts from the rest of the social media platforms. The model will be retrained using this data, because they are domain-specific (*these tweets and comments were made on Indian prime minister Narendra Modi and other leaders as well as people's opinion towards the next prime minister of the nation, in the context of India's elections*),[69] and the comparison would be fairer than using our domain-specific models.

At first, X posts from both sources have been isolated, and concatenated to have a unified dataset for training. The rest of the models' configuration remained the same as in this final version of the Sentiment Analysis module. The performance metrics are depicted in Table 18.

Table 18: Performance Metrics of Model trained on X Posts and Evaluated on Other Social Media

Social media platform Accuracy	Accuracy	Recall	Precision	F1
X	96.2%	96.1%	95.6%	95.8%
Other (Reddit, Facebook, Instagram)	92.6%	92.3%	91.6%	92.0%

It can be observed that there is some decrease in almost all metrics of the order of 4 units, however, this deviation is expected and acceptable. The primary concern when predicting posts from other social media platforms is the length of the posts. For example, in this case, where the model was trained on shorter sequences (X posts), so when longer texts are encountered (i.e., the comments of Reddit), the excess content is truncated, leading to potential loss of valuable information. Moreover, during the data collection process datasets from comments or headlines news have been identified to make the module's model more generic, in terms of social media platforms. Having said this, it can be safely argued that the Sentiment Analysis model is social media platform agnostic.

5.2.11.3 Specificity vs. Accuracy

As reported in the previous related deliverable (Deliverable 3.1), there is a considerable trade-off between specificity, that is, having a model that generalises as much as possible, and accuracy, having a model with the most accurate results. In the 2nd version of the module, this trade-off has been carefully balanced by selecting

[67] 'Twitter and Reddit Sentimental Analysis Dataset,' 2021. https://www.kaggle.com/datasets/cosmos98/twitter-and-reddit-sentimental-analysis-dataset

[68] 'Social Media Sentiments Analysis Dataset,' 2023. https://www.kaggle.com/datasets/kashishparmar02/social-media-sentiments-analysis-dataset

[69] 'Twitter and Reddit Sentimental Analysis Dataset,' 2021. https://www.kaggle.com/datasets/cosmos98/twitter-and-reddit-sentimental-analysis-dataset

domain-specific data, while not overly narrow to the use cases as to avoid the module's models being inflexible.

To showcase the balance between specificity and accuracy, the metrics of a model trained only on COVID-related topics (using the COVIDSenti dataset only) are presented in comparison with the health-related model present in the second version, which captures the sentiment for other health-related topics such as vaccination, medical drugs and health in general. The metrics for the released health-related model and the COVID specific model are depicted in Table 19 and Table 20, correspondingly. It is evident that the more specific the model is, the more accurate and robust it becomes.

Table 19: Evaluation Metrics of Public Health-Related Model

Eval Accuracy	Eval F1	Eval Loss	Eval Recall	Eval Precision	Train Accuracy	Train F1	Train Loss	Train Recall	Train Precision
0.908	0.847	0.429	0.870	0.827	0.944	0.939	0.376	0.945	0.933

Table 20: Evaluation Metrics of Model Trained only on COVIDSenti Dataset

Eval Accuracy	Eval F1	Eval Loss	Eval Recall	Eval Precision	Train Accuracy	Train F1	Train Loss	Train Recall	Train Precision
0.952	0.920	0.390	0.923	0.918	0.971	0.953	0.464	0.959	0.946[70]

5.2.11.4 Multilingual Posts

Language limitations are a common challenge in nearly all text-related tasks. The scarcity of high-quality multilingual datasets, particularly in specific domains like health and politics, makes training robust multilingual models difficult.

Although the current models of the Sentiment Analysis module are trained on English posts, the workflow supports other languages (German, French, Swedish, Finish and Spanish) through integrated translation mechanisms, ensuring sentiment prediction for non-English posts. Specifically, we use translator models developed by the Language Technology Research Group at the University of Helsinki, which can be accessed through Hugging Face's relevant hub[71].

Advanced multilingual approaches include training cross-lingual language model – RoBERTa (XLM-R)[72] – or multilingual BERT (mBERT)[73] models on multilingual datasets, if available, or employing zero-shot learning for cross-language sentiment prediction without translation.[74] These strategies would further enhance the adaptability and performance of the module across multiple languages serving as a compelling area for future research improvements.

5.3 Achievement of KPIs, KRs, and TOTs

According to the GA, a "FERMI sentiment analysis module (KR3.3)", which is further specified as a set of "intelligent and accurate sentiment analysis facilitators for fake news (KR2.6)",[75] was to be developed, which is fully in line with the module explained above. Just as well, the primary KPI assigned to KR2.6 and KR3.3 was the production a model with accuracy higher than 90% and reaching TRL-6; that is, by assessing its

[71] Helsinki-NLP. *Helsinki-NLP Models on Hugging Face*. https://huggingface.co/Helsinki-NLP

[72] Conneau, A., et al., 'Unsupervised Cross-lingual Representation Learning at Scale,' *arXiv preprint*, 2020.

[73] Devlin, J., et al., 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,' *arXiv preprint*, 2019. https://doi.org/10.48550/arXiv.1810.04805

[74] Wang, J., et al., 'Zero-Shot Cross-Lingual Summarization via Large Language Models,' *arXiv preprint*, 2023. https://doi.org/10.48550/arXiv.2302.14229

[75] Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021, PART B, p.4.

functionality and reliability in the operational environment of LEAs, thereby contributing to the optimisation and enhancement of ITML's 3ACEs toolkit.[76]

In the 1st version of the module (reported in Deliverable 3.1), due to considerable challenges and limitations, which were identified and analysed, the offered model had an accuracy of almost 70%. In the current development phase, a model with the evaluation metrics depicted in Table 21 has been produced**. It can be observed that this model reaches and, in some cases, surpasses the KPI target**. The model is specialised in health-related topics in accordance with the identified challenge of domain-specific data.

Table 21: Accuracy of Sentiment Analysis Module Best Performing Model

Train Accuracy	Eval Accuracy	Test Accuracy
94.4%	90.8%	89.7%

Even though the politics-related model does not reach the KPI threshold, its performance metrics have increased considerably from the previous version, as depicted in Table 22.

Table 22: Accuracy of the Politics-Related Model

Train Accuracy	Eval Accuracy	Test Accuracy
87.8%	84.5%	82.4%

Moreover, all models provided from the module have achieved and, in some cases, greatly exceeded the technological targets of Recall and F1 score as shown in Table 23 and Table 24, which were set to more than 60%. These technical targets were selected to better evaluate the models' ability to generalise, as these metrics are more robust to class imbalance.

Table 23: Recall Metric of the Sentiment Analysis Module's Models

Model	Train Recall	Eval Recall	Test Recall
General	78.7%	73.5%	72.4%
Health	94.5%	87.0%	86.2%
Politics	86.1%	81.9%	80.8%

Table 24: F1 Metric of the Sentiment Analysis Module's Models

Model	Train F1	Eval F1	Test F1
General	76.4%	71.4%	67.0%
Health	93.9%	84.7%	83.2%
Politics	85.7%	81.6%	79.8%

5.4 Versatility to Changing End-User Needs

Deliverable 3.1 highlighted that, at the time of its submission, the Sentiment Analysis module required further improvements to meet the performance target of achieving an accuracy of > 90%, as specified in the GA. Particular emphasis was placed on ensuring the tool's effectiveness in addressing the specific use cases outlined. Additionally, the module had yet to reach TRL-6, the technical maturity benchmark mandated by the GA. The identified limitations and objectives in the 1st version of the module, played a pivotal role in shaping the enhancement strategies undertaken for the module's development.

To reach the targeted KPI, domain specific datasets aligned to the identified use cases, were collected. However, to balance the trade-off between specificity and accuracy, and avoid limiting the module's capabilities to only the initial use cases (COVID-19, left-/right-wing extremism), the model's training data were expanded, to encompass more general domains like health and politics. **This approach ensures the**

[76] The 3ACEs toolkit can be accessed at https://www.itml.gr/products/analytics-as-a-service.

module's generalisability and transforms it to a versatile tool that can address a range of topics without requiring extensive retraining. The focus on broad thematic areas enables the module to remain relevant and applicable to evolving user needs, even as new societal and political challenges arise. **As a result, the Sentiment Analysis module provides at least one model with accuracy 90.8%, reaching the targeted KPI of accuracy > 90%**.

Additionally, even though the model has been primarily tested on X, the module's architecture is not limited to this platform. To ensure platform-agnostic functionality, datasets comprising diverse types of texts, such as headlines, and comments were incorporated. Furthermore, the model's performance on social media content beyond X was evaluated. Said testing demonstrated that while minor performance drops occur when applied to other platforms, the overall accuracy remains within acceptable limits, showcasing its readiness for broader use.

Just as well, feedback from pilot-users has been integrated into the development process. For instance, concerns about evaluating the impact of D&FN campaigns on public opinion have led to a clearer framing of the module's objectives. The sentiment analysis contributes to evaluating the impact of disinformation campaigns by identifying and analysing shifts in sentiment trends. To elaborate more on this, sentiment analysis is referring to the graph created by the Spread Analyser, which represents a rather small subset of public opinion. The bigger this graph is, the more likely it is to have a better grasp of the overall opinion of the users about the initial tweet provided by the LEA end-user. The impact mentioned in the requirement, in the context of sentiment, can be viewed by the end-user in the corresponding graph and provide information about the sentiment trends and shifts over time, as well as the overall sentiment of the social network captured by the tool. The module evaluates reactions within this medium rather than comprehensive public opinion metrics. In accordance with the clear legal and ethical limitations placed on FERMI's research (as analysed and summarised in the ethics deliverables D7.1-D7.3), such indiscriminate social media data processing is not permissible.

5.5 Sentiment Analysis Module Summary

The Sentiment Analysis module, which analyses D&FN circulating on social media, provides end-users with a perception of the emotional tone that characterises the content of posts. Importantly, the Sentiment Analysis module accomplishes said analysis utilising BERT, as defined in the GA and ensures the anonymisation of the posts, deletion of links, and replacing of emoji characters with corresponding text/keyword, protecting the privacy of European citizens. The Sentiment Analysis module provides end-users with a holistic understanding of the sentiment behind the network sharing the D&FN they are investigating. Moreover, the module offers models with very high-performance metrics and at least one model with accuracy higher than 90%, fully achieving the KPI target.

After the conducted literature review, the development began with evaluating two approaches, the fine-tuning of a Bert-based model and the feature extraction approach along with a bidirectional LSTM classifier. Additionally, methodological choices such as different cleaning pipelines and strategies for tackling imbalanced datasets were evaluated and the best configuration was selected. The challenges and limitations identified in the initial iteration (see Deliverable 3.1) drove the actions towards the advancements of the 2nd and final version of this module.

Extensive research was conducted to identify and evaluate datasets that were domain-specific to the project's use cases, while taking into consideration the balance between specificity and accuracy, to avoid the creation of models that were too specific and inflexible to changing end-user needs. The effect of this trade-off has been also evaluated by comparing a COVID-related model to a more general health-related one. Additionally, the developers made the module's model social media platform agnostic, by enriching the training datasets with records from sources other than X and evaluating the effect of using different social media posts during training and inference. Finally, some initial research for multilingual approaches has been conducted setting the path for future advancements following the completion of the FERMI platform. Regarding the pilots' feedback, the Sentiment Analysis module satisfies the requirement to provide users with the ability to analyse the sentiment polarity of social media posts related to the identified pieces of disinformation. On the other hand, it was argued that the analysed social media posts could not fully resemble the public opinion in general but can provide meaningful insights into it. The pilots' participants' concern is valid, as social media data is inherently biased and the analysed ones reflect only a subset of the general public opinion, often those of more active or vocal users. However, we need to clarify that the relevant user requirement elicited from the practitioner interviews and survey in WP2 can, at least in the context of the Sentiment Analysis module, only alludes to the ability to evaluate the sentiment impact of disinformation campaigns on public opinion rather than capturing or quantifying public opinion itself (User Requirement 29, “The user is able to evaluate the impact of disinformation campaigns on public opinion”). On this basis, the Sentiment Analysis module contributes to evaluating the impact of disinformation campaigns, in the context of sentiment, by identifying and analysing shifts in sentiment trends. This is achieved through the provided sentiment over time figure, which **conveys information about the sentiment trends and shifts over time, as well as the overall sentiment of the social network captured by the tool**. In any case, it is important to recognise this limitation of the tool and accept that while social media posts cannot replace comprehensive public opinion studies, they provide valuable insights into specific narratives, influencers, and trends, particularly in relation to disinformation.

Despite the limitations mentioned above, the detailed analyses and strategic actions performed during the second period of the project, together strengthen the effectiveness and reliability of the Sentiment Analysis module, making it an important and reliable part of the larger FERMI platform.

6 Integration with Tasks 3.3 & 3.5

Tasks 3.3 and T3.5 operate within the greater FERMI platform and rely on the flow of data analysis that begins with the Spread Analyser. However, most important to their operation is the Dynamic Flows Modeler, whose output is used as the input for T3.5. The aim of these two tasks, the Behaviour Profiler & Socioeconomic Analyser and the Community Resilience Management Modeler focuses on assessing the implications of the end-user provided D&FN in terms of severity and likelihood and providing potential countermeasures the end-user can employ. In section 6, the technologies will be briefly explained, with a majority of the attention placed on the integration between them and the technologies central to D3.2 (i.e., T3.1, T3.2, T3.4, T3.6). For greater coverage of T3.3 and T3.5's development and adherence to the GA, one should refer to D3.4: FERMI Behavioural Analyses and Community Resilience Facilitators Package – 2nd version.

6.1 Task 3.5 – Behaviour Profiler and Socioeconomic Analyser

The Behaviour Profiler consists of two sub-tracks, the necessity to understand the number of offline crime occurrences following an online D&FN campaign, and an analysis of how the resident of a specific country may react to the same online D&FN campaign, considering factors including media literacy and information consumption behaviour. More specifically, the Behaviour Profiler, in its role as one of the valid predictors, entails the need to foresee the number of offline crime occurrences, which is provided by the Dynamic Flows Modeler. Using past crime and past D&FN, the Dynamic Flows Modeler helps to estimate the change in crime occurrences for four crime types, in the end-user's requested NUTS2 region.

> As described in the GA, the Behaviour Profiler, specifically its *country profiles*, relates to politically motivated extremism['s]… impact on society… [with an aim to] determine effects of online propaganda on offline actions. In this respect, the degree of media literacy may tend to correspond to the degree of resilience of a society. The means of information and news consumption is a first indicator for the assessment of media literacy. Factors such as the type of source, the 'general' assessment of the medium, the level of trust (if feasible) and differentiation by age groups (demographics) play an important role. Based on secondary literature, an analysis of the media literacy of certain countries will be conducted, considering the factors mentioned above. This preliminary work allows behavioural profiles to be better differentiated and classified.[77]

As for the **Socioeconomic Analyser**, it **is concerned with the connection between D&FN and crime, specifically said crime's effects on economically measurable variables**, such as gross domestic product (GDP) per inhabitant. According to the GA, the Socioeconomic Analyser, through "applying econometric methods" will reveal "the effects of radicalization and extremism… reflected in financial terms to quantify the costs of… [D&FN's] negative effects (based on data availability in the respective country/region) for the society."[78] **The intuition behind this originates from academic research** that built the theoretical foundation for looking at possible effects of violent political extremism on economic variables. It has been proposed that **violent political extremism leads to a loss in social welfare** via different channels. These may concern the deterrence of investors, the influence on political decisions and instructional output as well as trade and further factors.[79]

The main model to explain economic costs by politically motivated crime is depicted in the following regression equation.

$$Prod_{rt} = \alpha + \beta_1 Ext + X_{rt} + v_r + \varepsilon_{rt}$$

Equation 11: Calculation of Economic Costs of Political Extremism

[77] 'Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021.

[78] Ibid.

[79] Ferguson, N., et al., 'Die Kosten des Extremismus,' *BIGS Standpunkt zivile Sicherheit*, 2019.

It explains the measurement of productivity for a given region (r) for a given time period (t). α is the constant for the regression, v_r is a vector of time-invariant region-specific properties. ε_{rt} is the error term for the regression. ***Ext*** **is the measurement of extremism in terms of crime** and β_1 gives the cost coefficient, i.e. how a single unit increase in crime will affect economic welfare. Lastly, X_{rt} describes a vector of control variables, such as size of the region. It is in this equation that the integration between technologies is most relevant. The number of ***Ext*, in this equation, is sourced from the Dynamic Flows Modeler**, which provides a level of offline crime occurrence following a D&FN event online.

6.2 Task 3.3 – Community Resilience Management Module

The Community Resilience Management Modeler and Disinformation Watch joint component aim to aid LEAs in prioritising the correct course of action for tackling D&FN-related crime. Considering the impact produced by a crime, the tool will output a ranking of countermeasures to tackle high-stakes D&FN events. This output results from a multi-criteria decision analysis that produces a decision model and has the LEA as the decision-maker. The decision model and subsequent additive model will consider the LEA consensual opinion to provide options for tackling disinformation. Achieving a consensus on what options to adopt will be obtained through a DELPHI study initiative. Furthermore, the decision model will consider predefined criteria, referring to a specific factor the decision-maker uses to evaluate and assess options under consideration (e.g., media literacy index, media trust index, thread of poverty or social exclusion, etc.).

The component will also **consider a previously made assessment of a D&FN event's impact, measured by the product of the likelihood and the severity of a particular crime occurring in a specific community**. Provided that the assessment has an index value indicating that an investigation is of high or extremely high impact on the community, the system will output a ranking of countermeasures specific to the instance of crime being investigated by the end-user. On the other hand, should the impact index be minimal or medium, the system will not output any countermeasures, and a message that no action is advisable will be provided to the end-user.

In this way, **LEAs will gain insight into whether to reallocate resources to counteract the D&FN under investigation or not**. The impact assessment of a D&FN event on the community in question is provided by the Behaviour Profiler & Socioeconomic Analyser, which, once given the potential number of crime occurrences by the Dynamic Flows Modeler, generates a measure of the D&FN in question's impact, in economic terms.

7 Conclusion

Deliverable 3.2, the technology facilitators package – 2nd version, presented the end-product versions of the technological components developed within the framework of the FERMI project for the FERMI platform. As a sequel deliverable, greater attention was paid to the steps taken after the submission of Deliverable 3.1 as well as the improvements made to address feedback from pilot-users and the project review report. Specifically, the technologies developed within T3.1 (the Dynamic Flows Modeler), T3.2 (the Spread Analyser), T3.4 (SL framework), and T3.6 (the Sentiment Analysis module) are reported on. For each technology, there is a dedicated section for a practical description of the components' function, a more nuanced technical description, the achievement of KPIs, KRs, and TOTs, as well as the versatility of the development to changing end-user needs.

The Dynamic Flows Modeler successfully evaluates the "the intensity of the relation between the spread of D&FN and offline crimes, the temporal patterns in the relation, [and] the spatial decay of the relation"[80] in its capacity to produce informed, accurate estimates for offline crime occurrences following a D&FN event online. The Dynamic Flows Modeler is supported by the completion of a first, functional SL framework that allows "for training Machine Learning models near to the data sources where they are generated,"[81] where the data sources are several, independent European LEAs. Given the SL infrastructure, said LEAs do not need to sacrifice any degree of privacy and data protection nor turnover any confidential information posing no risk to individuals' personal data for which they act as controller. The Spread Analyser is a powerful technology capable of taking "as input news already classified as [D&FN] and… [mapping] this news to their main actors/accounts which are responsible for creating and spreading the [D&FN] across the network."[82] The Spread Analyser, adhering to the GA's commitments, can classify if the identified actors/accounts are physical persons or bots and assign an influence index to their role/power over the network. The Sentiment Analysis module, which analyses D&FN, specifically in social media posts, to provide end-users a perception of the emotional tone in said posts' content, exploits BERT, as promised in the GA. keywording sum, the Sentiment Analysis module provides end-users with a wholistic understanding of the sentiment behind the network sharing the D&FN they are investigating.

The analysis of social media messages is preceded by ensuring the anonymisation of the posts, deletion of links, and replacing of emoji characters with corresponding text/keyword. As previously mentioned, **components developed with the FERMI project have shifted to being social media agnostic**, meaning that despite certain social media networks were relied on for testing and development, the tools have been adapted such that **they may be adjusted and applied to any social media platform, should access become available**.

T3.3 and T3.5, further offerings by the FERMI platform, are examined in depth in Deliverable 3.3 and Deliverable 3.4. Deliverable 3.2 mentions, with brevity, their functionality and how they are integrated with the technologies in focus. Specifically, how the Dynamic Flows Modeler provides the estimated number of crime occurrences, from which the Behaviour Profiler and Socioeconomic Analyser produce an understanding of the potential harm, in economic terms, deriving from the D&FN provided by the end-user. The Community Resilience Management Modeler, then, informs end-users as to what potential counter measures may be taken.

Given the SOTA, LEAs are seeking means by which they can understand the threat of D&FN and build a broader intelligence picture for D&FN campaigns. The aforementioned components provide needed insights, with which LEAs can make better-informed choices regarding the distribution of resources and type of response. The components developed within the FERMI project are pioneering solutions against the threat of D&FN, **offering to LEAs new degree of analytical insights through a direct linkage with social media platforms**.

[80] 'Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,' *European Research Executive Agency*, 2021.
[81] Ibid.
[82] Ibid.

References

Abdeljaber, O., Avci, O., Gabbouj, M., Ince, T., Inman, D.J., & Kiranyaz, S., '1D Convolutional Neural Networks and Applications: a Survey,' *Mechanical Systems and Signal Processing*, Vol. 151, No 107398, 2021.

Antypas, D., Preece, A., Camacho-Collados, J., 'Negativity spreads faster: A large-scale multilingual twitter analysis on the role of sentiment in political communication,' *Online Social Networks and Media,* Vol. 33, 2023. https://doi.org/10.1016/j.osnem.2023.100242.

Aziani, A., Lo Giudice, M.V., Shadman Yazdi, A., 'Conspiracy to Commit: Information Pollution, Artificial Intelligence, and Real-World Hate Crimes,' *European Journal of Criminal Policy and Research,* 2025 https://doi.org/10.1007/s10610-025-09629-w.

Aziani, A., Lo Giudice, M.V., Shadman Yazdi, A., 'Conspiracy to Commit: Information Pollution, Artificial Intelligence, and Real-World Hate Crimes,' *Eurocrim2025 – European Society of Criminology*, Athens, Greece, 2025.

Aziani, A., 'Exploring the Nexus between Information Pollution and Offline Criminal Events,' *American Society of Criminology 78th Annual Meeting*, Philadelphia, Pennsylvania, 2023.

Baraniak, K., Marcin, S., 'SEN - Sentiment analysis of Entities in News headlines,' *Zenodo,* Vol. 192, pp. 3627-3636, 2021.

Belda, S., Dhar, S., Ferrandiz, J.M., Guessoum, S., Heinkelmann, R., Modiri, S., Raut, S., & Schuh, H., 'The Short-Term Prediction of Length of Day Using 1D Convolutional Neural Networks (1D CNN),' *Sensors*, Vol. 22, No 9517, 2022.

Berkeley Vision and Learning Center, 'Releases of Caffe: A Fast Open Framework for Deep Learning', *GitHub repository*, n.d. https://github.com/BVLC/caffe/releases.

Beutel, Daniel J., et al., 'Flower: A Friendly Federated Learning Research Framework', *arXiv preprint arXiv:2007.14390*, 2020. https://arxiv.org/abs/2007.14390.

Botticher, A., 'Towards Academic Consensus Definitions of Radicalism and Extremism' *Perspective Terror*, Vol. 11, No 4, 2017, pp. 73 – 77.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V., 'Unsupervised Cross-lingual Representation Learning at Scale,' *arXiv preprint*, 2020. https://doi.org/10.48550/arXiv.1911.02116.

'Covid-19 Vaccine Tweets with Sentiment Annotation,' n.d. https://www.kaggle.com/datasets/datasciencetool/covid19-vaccine-tweets-with-sentiment-annotation.

Deliverable 5.2 FERMI 1st execution reports

Devlin, J., Chang, M., Lee, K., Toutanova, K., 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,' *arXiv preprint*, 2019. https://doi.org/10.48550/arXiv.1810.04805.

'ECE143-Political-Sentiment-Analysis,' n.d. https://github.com/akashboghani/ECE143-Political-Sentiment-Analysis/tree/master/data.

Eurostat. (2019-2025). Eurostat Database. https://ec.europa.eu/eurostat/web/main/data/database.

Fergusen, N., Rieckmann, J., Stuchtey, T., ‘Die Kosten des Extremismus,’ *BIGS Standpunkt zivile Sicherheit,* Vol. 9, 2019. https://www.bigs-potsdam.org/app/uploads/2020/06/BIGS-Standpunkt_Nr.-9-2019_Kosten-des-Extremismus_WEB.pdf.

‘First GOP Debate Twitter Sentiment,’ n.d. https://www.kaggle.com/datasets/crowdflower/first-gop-debate-twitter-sentiment?select=Sentiment.csv.

Flower.ai, 'Comparison of Federated Learning Frameworks', Flower.ai Blog, available at: https://flower.ai/blog/2024-07-22-fl-frameworks-comparison/.

‘General Project Review Consolidated Report (HE) - Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01’, *European Research Executive Agency,* 2024

‘Grant Agreement: Project 101073980 – FERMI – HORIZON-CL3-2021-FCT-01,*’ European Research Executive Agency*, 2021.

Hassija, V., Chamola, V., Mahapatra, A., et al., ‘Interpreting Black-Box Models: a Review on Explainable Artificial Intelligence,’ *Cognitive Computing*, Vol. 16, 2024, pp. 45 – 74.

Hawley, G., *Making Sense of the Alt-Right*, New York Chichester, West Sussex: Columbia University Press, 2017, pp. 115-138. https://doi.org/10.7312/hawl18512-007.

‘Healthcare Related Tweets for Sentiment Analysis,’ *Omdena*, n.d. https://datasets.omdena.com/dataset/healthcare-related-tweets-for-sentiment-analysis.

‘Helsinki-NLP Models on Hugging Face,’ *Helsinki-NLP*, n.d. https://huggingface.co/Helsinki-NLP.

Katarzyna, B., Sydow, M., ‘A Dataset for Sentiment Analysis of Entities in News Headlines,’ *Procedia Computer Science*, Vol. 192, pp. 3527 – 3636, 2021. https://doi.org/10.1016/j.procs.2021.09.136

Li, Y., Ma, W., Chen, C., et al., ‘A Survey on Dropout Methods and Experimental Verification in Recommendation,’ *IEEE Transactions on Knowledge and Data Engineering*, Vol. 35, No 7, 2023, pp. 6595 – 6615.

Lo Giudice, M.V., Aziani, A., Shadman Yazdi, A., ‘Conspiracy to Commit: Information Pollution, Artificial Intelligence, and Real-World Hate Crimes,’ *American Society of Criminology 79th Annual Meeting,* San Francisco, California, 2024.

Lo Giudice, M.V., Dugato, M., Favarin, S., ‘Disinformation and Crime: The Nexus Between Online Disinformation and Offline Crime,’ *Eurocrim2025 – European Society of Criminology,* Athens, Greece, 2025.

Lo Giudice, M.V., Shadman Yazdi, A., Aziani, A., ‘Informative (Dis)Information: Exploring the Correlation Between Social Media Disinformation Campaigns and Real-World Criminal Activity,’ In proceedings of *2024 5th International Conference in Electronic Engineering, Information Technology & Education (EEITE)*, Chania, Greece, 2024.

Naseem, U., Razzak, I., Khushi, M., et al., “COVIDSenti: A Large-Scale Benchmark Twitter Data Set for COVID-19 Sentiment Analysis,” *IEEE Transactions on Computational Social Systems*, 2021. https://doi.org/10.1109/TCSS.2021.3051189.

‘Number of Registered Mastodon Useres Worldwide as of March 2023,’ *Statistica*, 2023, https://www.statista.com/statistics/1376022/global-registered-mastodon-users/.

‘Political Sentiment Analysis,’ n.d. https://www.kaggle.com/datasets/subhajournal/political-sentiment-analysis.

Restack, 'PyTorch vs TensorFlow vs Keras vs Theano vs Caffe: Detailed Comparison,' *Restack.io*, https://www.restack.io/p/pytorch-answer-vs-tensorflow-vs-keras-vs-theano-vs-caffe.

Riedel, P., Schick, L., von Schwerin, R. et al. ‘Comparative analysis of open-source federated learning frameworks - a literature-based survey and review.’ *Int. J. Mach. Learn. & Cyber*. Vol. 15, 2024, pp. 5257–5278. https://doi.org/10.1007/s13042-024-02234-z

‘Right-wing Extremism,’ *Bundesamt für Verfassungsschutz*, n.d.

‘RogerKam/roberta_fine_tuned_sentiment_sst3,’ n.d. https://huggingface.co/RogerKam/roberta_fine_tuned_sentiment_sst3.

Rosenthal, S., Farra, N., Nakov, ‘SemEval-2017 task 4: Sentiment analysis in Twitter,’ in proceedings of *the 11th International Workshop on Semantic Evaluation*, Vancouver, Canada, 2017. https://doi.org/10.18653/v1/S17-2088.

‘Sentiment Analysis for Medical Drugs,’ n.d. https://www.kaggle.com/datasets/arbazkhan971/analyticvidhyadatasetsentiment/data.

‘Social Media Sentiments Analysis Dataset,’ 2023. https://www.kaggle.com/datasets/kashishparmar02/social-media-sentiments-analysis-dataset.

‘TextBlob: Simplified Text Processing,’ n.d. https://textblob.readthedocs.io/en/dev/.

Torregrossa, J., Bello-Orgaz, G., Camacho, D., Del Ser, J., & Martinez-Camara, E., ‘A Survey on Extremism Analysis using Natural Language Processing: Definitions, Literature Review, Trends and Challenges,’ *Journal of Ambient Intelligence and Humanized Computing*, Vol. 14, 2022, pp. 9869–9905.

‘Twitter and Reddit Sentimental analysis Dataset,’ 2021. https://www.kaggle.com/datasets/cosmos98/twitter-and-reddit-sentimental-analysis-dataset.

‘Number of registered Mastodon users worldwide as of March 2023’, *Statista*, March 2023, May 2025, https://www.statista.com/statistics/1376022/global-registered-mastodon-users/

Wang, J., Liang, Y., Meng, F., Zou, B., Li, Z., Qu, J., Zhou, J. ‘Zero-Shot Cross-Lingual Summarization via Large Language Models,’ *arXiv preprint*, 2023. https://doi.org/10.48550/arXiv.2302.14229.

Wang, Y., Klabjan, D., & Pei, J., 'Prediction of crime occurrence from multi-modal data using deep learning,' *PLoS ONE*, Vol. 12, No 4, 2017.

Annex A Sentiment Analysis Module Generated X Posts

Table 25: Original negative tweets of "Healthcare Related Tweets for Sentiment Analysis" dataset, which were used in the prompt given as input to ChatGPT to generate 60 more tweets of the same class

Original Negative Tweets
These headaches everyday and high blood pressure is really killing me ðŸ˜ðŸ˜
@rebecca_sissons High blood pressure ðŸ˜¬
The world hopes that the cancer surgery of #Putin is a spectacular failure. #Russia #RussianWarCrimes #Scumbag #Murderer #ActualNazi https://t.co/4gIzZXDou7
like iâ€™m in PAIN, bad side effect from the surgery
@goodkinggabriel Eye surgery. Went terrible wrong.
NO ICU NO Inpatient NO Emergency Room Oklahoma Hospitals are in DIRE CRISIS. PATIENT CARE VERY LIMITED. Silence from Top Officials continue. #FactCheck
something weirdly sinister about playing â€˜only the good die youngâ€™ in an urgent care waiting room
cried in the urgent care examination room, very cancer rising of me lol Iâ€™m embarrassed
urgent care might be worse than the damn emergency room atp.
being stuck in an urgent care waiting room with no Instagram is PAINFUL and should be ILLEGAL
Waiting in an exam room at an urgent care and appreciating the music choice. Shawty did, in fact, get low low low low low low low low. With illness. https://t.co/Jas2TcMutD
@Myshiloh Crap hospital emergency room? I hope everything is ok ðŸ˜ž
find me in the emergency room in the hospital cuz i got a very severe fatal injury from kztrâ€™s cuteness attack
@killpundit Yes. Hospital. Emergency room. Scary needles. Tooth pain.
Icb am in the hospital in the emergency room and am still streaming ðŸ˜ðŸ˜ðŸ˜ðŸ˜ðŸ˜ðŸ˜ðŸ˜ðŸ˜ðŸ˜-ðŸ˜ðŸ˜ðŸ˜ðŸ˜ðŸ˜
@fox8news Suicide if it was! In the emergency room of a hospital probably assisted suicide! Can anyone say murder?
sobs in a hospital emergency room https://t.co/L4aP5HEMgA
Hospital and Emergency Room Fraud - Contact Us Today to Report Fraud https://t.co/QOFCIfnBZQ
Damn virus this morning my mother has been admitted to the emergency room, who tells me that that does not exist ðŸ˜¡ Of course there is, damn virus is very bad ðŸ˜¢ðŸ˜¢ðŸ˜¢ðŸ˜¢ðŸ˜¢ðŸ˜¢ I have no symptoms, I have nothing I want to be ðŸ˜¢ðŸ˜¢ðŸ˜¢ and for my mother to leave the hospital ðŸ˜¢ðŸ˜¢ðŸ˜¢ðŸ˜¢ðŸ˜¢ðŸ˜¢
Lag is sick... Time to treat Lag!
Oh shit am I a critical care nurse now?! ðŸ˜±ðŸ˜³ https://t.co/tPY7NuueCY
@gommmmh :(please take care this is so gross
with tears can relieved your pain.
#Paediatric Trauma &amp; Depression A traumatic incident is a distressing, dangerous, or stressful event that puts a child's life or body reliability in jeopardy. It can be traumatic to see a horrific event that threatens a loved one's life or physical security. https://t.co/x1eCtxtmof
A shot to kill the pain. A drink to drown the shame. No matter what I do, it always hurts the same
@stand_for_all This is called fright response to trauma we are traumatised

Depression is a mental health problem. It can be life threatening. #MentalHealthAwarenessWeek
RT @t33n4g3_d1rtb4g: mental illness is never an excuse for abuse.
health violations? no thanks, im very healthy
Absolutely SICK!! https://t.co/LPSLr6VKOX
@Saadia___M @Lynxanon No health care, no sex.
Looks like new strains evade vaccines, but vaccination still gives significant protection against the worst outcomes - death and critical illness. 2/ https://t.co/Ca0bBlxqb8
@BlueF0x3 I feel like shit, covid is kicking my ass rn. I am tired, in pain, it hurts to cough. so ya... I feel terrible
Now concerned about a Recession -&gt; mental health crisis. Depression, anxiety, panic, or problems with drug use rise when less ppl can afford therapy.
RT @theLaurenMarl: Death is not a solution to foster care. Death is not a solution to abuse. Death is not a solution to rape. Death is notâ€¦
RT @BustyThiccOni: Limits: Gore Horror Death Vore Scat Diapers Rimjobs Toilet related (Or any dirty places) Severe pain.
Emergency room visits aren't fun
im currently having a really painful body pain but HELL YEAH THEY'RE HAVING AN OFF COLLAB AGAIN ðŸ˜ðŸ˜ðŸ˜ðŸ˜
@astrotigre @LilyPichu That shit can cure cancer
1 year since surgery and shit bitch i lost 110lbs
@_slimarella_ pain index 3/10 itâ€™s more uncomfortable than painful.
@Lebogangsheldon Pain is temporary
I feel frightened and betrayed. Will I or my family get sick go bankrupt, die without affordable health insurance?#HealthcareBill
@NickForVA Prosecute the doctor for maiming, child abuse, assault with deadly weapon, med malpractice.
but filmaking is a major pain in the ass
Woke up sick af. Allergies whooping my ass
Having a sore throat is really the worst. Every second is pain
Calculus is pain.
@hippydog444 @ananavarro I agree sometimes the underlying physical illness and pain outweigh the pain suicide leaves her family! #notifwhen
ÙˆØ±Ø¨ÙŠ in serious pain ðŸ’”ðŸ’”ðŸ’”ðŸ’” https://t.co/aNam81cUT2

my side was the worst pain ever omg https://t.co/mt79SPH8Cu
RT @WAI_Alzheimers: "Long-term stress of any kind, including caregiver stress, can lead to serious health problems, including depression anâ€¦
@PeanutsHatesMe @KellyQuilt i'm allergic to excessive labor, and pain. pain is big one. i don't like it. like, at all. sometimes i moan and groan and scream, it's horrible. just freaking horrible.
"there almost no pain right now. but when i got hurt it was scary" â˜¹ðŸ˜ https://t.co/7oGHcjcmqA
Pain is dumb, pain medication is worse, and food is also overrated
good morning being sick is quite literally the worst feeling ever
Short time pain is better than regret.
pain killers, sleeping pills and sinus tablets are addictive as hell.
RT @NorthS1de_Jay: Health care in this country is the CRAZIEST scam I hate this country and this life so bad ðŸ˜ðŸ˜ðŸ˜ðŸ˜ðŸ˜
@iowademocrats or death care?
I. AM. SICK. AND. TIRED!!!!!!!! STOP IT!!!!!!!!!!!!! NOW!!!!!!!!!!!!!!! @itskerrii @21Blackking @QRich @MicahhParsons11 https://t.co/5pmsWaIeNm
@ChelleMarie393 Reported and block. They are sick.
For the pain pls. https://t.co/SXiyY3tadN
Alcohol always numb the pain!
being sick all alone is a different type of pain
RT @jacamprodon: ECT saves lives: Risk of suicide death following electroconvulsive therapy treatment for depression: a propensity score-wâ€¦
@MrAndyNgo The poor starving masses stealing food, medicine and diapers. @AOC
RT @KathleenKokolus: Crisis in Ukraine Disrupts Delivery of Critical Cancer Care | Cancer @Labroots https://t.co/zxBP3XbrLK

Table 26: Generated tweets for slightly balancing the negative class of "Healthcare Related Tweets for Sentiment Analysis" dataset

Generated Tweets
These migraines every day are driving me crazy 😭.
Feeling awful after my surgery, the pain is unbearable.
Eye operation went horribly wrong, so disappointed.
Something eerie about playing sad songs in a hospital waiting room.
Broke down crying in the urgent care room, so embarrassing.
Waiting in urgent care is worse than the emergency room.
Stuck in an urgent care waiting room, no WiFi, this is torture.

Hospital emergency room is a nightmare, hope everything's okay 😞 .
I'm in the ER with a severe injury, this is terrifying.
Emergency room visit, painful needles and all.
In the ER, contemplating life, this feels like a slow death.
Crying in a hospital emergency room, this is awful.
My mom got admitted to the ER this morning, damn virus 😠 😢 .
Am I really working in critical care now? This is surreal 😱 .
Trauma responses are no joke, we are all traumatized here.
Depression is serious, it can be life-threatening.
No excuse for abuse, even if you're mentally ill.
Feeling absolutely sick to my stomach 😠 .
Covid is kicking my butt, feeling terrible and in pain.
Recession worries lead to a mental health crisis, anxiety is up.
Limits include gore, horror, and severe pain, can't handle it.
Visiting the emergency room is never a fun experience.
My body is in so much pain right now, this is hell 😭 .
Pain is temporary, but it feels endless.
Scared and betrayed, will my family survive without health insurance?
Woke up feeling sick, allergies are the worst.
Sore throat is killing me, every second is painful.
Calculus feels like torture.
My side hurts so much, worst pain ever.
The crisis in Ukraine is disrupting critical cancer care, so tragic.
These headaches and high blood pressure are unbearable 😩 .
I hate how the world is right now, everything feels so hopeless.
My surgery had awful side effects, I'm in so much pain.
The urgent care room is like a scene from a horror movie.
Crying in the hospital again, I can't handle this.
Urgent care is just as bad as the ER, if not worse.
Being in an urgent care waiting room feels like torture.
The music in this hospital is making my anxiety worse.
I'm terrified, waiting in the ER for test results.
The hospital emergency room is chaos, I'm so scared.
Stuck in the ER with a severe injury, this is the worst.
Feeling utterly helpless in the emergency room 😢 .
My mother is in the ER because of this horrible virus 😠 .
I can't believe I'm working in critical care now, it's terrifying.
Trauma leaves scars that never heal, feeling broken.
Depression feels like a never-ending black hole.
Mental illness is never an excuse for being abusive.
I'm feeling so sick and tired of everything 😠 .
This flu is killing me, I feel absolutely terrible.
Economic downturns always trigger my anxiety and depression.
Severe pain is something I can't handle, it's overwhelming.
Hospital visits always bring out the worst fears in me.
My body aches all over, I feel like I'm dying 😢 .

Pain might be temporary, but it feels eternal right now.
I'm scared we'll go bankrupt without proper health insurance.
Allergies are hitting me hard, feeling miserable.
A sore throat is making every moment agony.
This math homework feels like pure torture.
Worst side pain ever, can't move without hurting.
The healthcare system is a nightmare, I'm so frustrated.