Crowd4SDG
Citizen Science for the Sustainable Development Goals

Deliverable 2.1

CS tools design and early prototype available

Deliverable identifier: D2.1

Due date: 30/04/2021

Justification for delay: due to exceptional activity as agreed with Project Officer

Document release date: 21/06/2021

Nature: Report

Dissemination Level: Public

Work Package: 2

Lead Beneficiary: CSIC

Contributing Beneficiaries: POLIMI, UNITAR, UNIGE

Document status: Final

Abstract:

One of the main objectives of the Crowd4SDG project is to create a toolkit for Citizen Science, the so-called Citizen Science Solution Kit, which incorporates Artificial Intelligence techniques. This document summarizes the work performed during the first year of the Crowd4SDG project regarding the design of the Citizen Science Solution Kit, and presents the functioning prototypes. Furthermore, the document also describes the main activities and achievements of the work package 2 *"Enhancing Citizen science tools and methodologies"*.

For more information on Crowd4SDG, please check: http://www.crowd4sdg.eu/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 872944.

Document history

	Name	Partner	Date
Authored by	Jesús Cerquides Bueno Oguz Mulayim Barbara Pernici Cinzia Cappiello Rosy Mondardini	CSIC CSIC POLIMI POLIMI CSCZ	22/10/2020
Edited by	Jesús Cerquides Bueno	CSIC	15/03/2021
Reviewed by	Laura Wirtavuori Romain Muller Camille Masselot Marc Santolini	CERN CERN UP UP	06/04/2021
Edited by	Jesús Cerquides Bueno Oguz Mulayim	CSIC	14/05/2021
Reviewed by	Jose Luis Fernandez-Marquez	UNIGE	16/06/2021
Edited by	Jesús Cerquides Bueno Oguz Mulayim	CSIC	17/06/2021
Approved by	François Grey Jose Luis Fernandez-Marquez	UNIGE	18/06/2021

Table of Contents

Project Partners

	Partner name	Acronym	Country
1 (COO)	Université de Genève	UNIGE	CH
2	European Organization for Nuclear Research	CERN	CH
3	Agencia Estatal Consejo Superior de Investigaciones Científicas	CSIC	ES
4	Politecnico di Milano	POLIMI	IT
5	United Nations Institute for Training and Research	UNITAR	CH
6	Université de Paris	UP	FR

Crowd4SDG in Brief

The 17 Sustainable Development Goals (SDGs), launched by the UN in 2015, are underpinned by over 160 concrete targets and over 230 measurable indicators. Some of these indicators initially had no established measurement methodology. For others, many countries do not have the data collection capacity. Measuring progress towards the SDGs is thus a challenge for most national statistical offices.

The goal of the Crowd4SDG project is to research the extent to which Citizen Science (CS) can provide an essential source of non-traditional data for tracking progress towards the SDGs, as well as the ability of CS to generate social innovations that enable such progress. Based on shared expertise in crowdsourcing for disaster response, the transdisciplinary Crowd4SDG consortium of six partners is focusing on SDG 13, Climate Action, to explore new ways of applying CS for monitoring the impacts of extreme climate events and strengthening the resilience of communities to climate related disasters.

To achieve this goal, Crowd4SDG is initiating research on the applications of artificial intelligence and machine learning to enhance CS and explore the use of social media and other non-traditional data sources for more effective monitoring of SDGs by citizens. Crowd4SDG is using direct channels through consortium partner UNITAR to provide National Statistical Offices (NSOs) with recommendations on best practices for generating and exploiting CS data for tracking the SDGs.

To this end, Crowd4SDG rigorously assesses the quality of the scientific knowledge and usefulness of practical innovations occurring when teams develop new CS projects focusing on climate action. This occurs through three annual challenge based innovation events, involving online and in-person coaching. A wide range of stakeholders, from the UN, governments, the private sector, NGOs, academia, innovation incubators and maker spaces are involved in advising the project and exploiting the scientific knowledge and technical innovations that it generates.

Crowd4SDG has six work packages. Besides Project Management (UNIGE) and Dissemination & Outreach (CERN), the project features work packages on: Enhancing CS Tools (CSIC, POLIMI) with AI and social media analysis features, to improve data quality and deliberation processes in CS; New Metrics for CS (UP), to track and improve innovation in CS project coaching events; Impact Assessment of CS (UNITAR) with a focus on the requirements of NSOs as end-users of CS data for SDG monitoring. At the core of the project is Project Deployment (UNIGE) based on a novel innovation cycle called GEAR (Gather, Evaluate, Accelerate, Refine), which runs once a year.

The GEAR cycles involve online selection and coaching of citizen-generated ideas for climate action, using the UNIGE Open Seventeen Challenge (O17). The most promising projects are accelerated during a two-week in-person Challenge-Based Innovation (CBI) course. Top projects receive further support at annual SDG conferences hosted at partner sites. GEAR cycles focus on specific aspects of Climate Action connected with other SDGs like Gender Equality.

Grant Agreement description of the deliverable

Parts in bold describe the deliverable content.

WP2 - Enhancing Citizen science tools and methodologies [Months: 1-36]
CSIC, UNIGE, POLIMI, UNITAR
The work in each of the research fields will evolve through 3 different phases: **(1) design and early prototyping**, (2) beta release and (3) final release. The results from each phase will be demonstrated in a public event at the closing ceremony of each GEAR cycle, i.e. at M11, M21, and M30.
The focus will be on enhancing existing open source Citizen Science tools (see Section 1.3.7.4) with AI components to work with large-scale datasets, improve data quality and speed up data analysis. The tools will be used during WP3 (specifically: Task 3.1) GEAR cycles in order to validate them, and support participants on the creation of CS projects.

D2.1 : CS tools design and early prototype available [12]
CS tools design and early prototype available. The demonstration of the early prototype will be included in this deliverable.

Purpose and scope of the deliverable

This deliverable contains a description of the works developed during the first year in Work Package 2 (WP2) “Enhancing Citizen science tools and methodologies” and a detailed description of the design and early prototype of the Citizen Science Solution Kit (CSSK) which is a result of the work package.

1. Introduction

This deliverable contains a description of the works developed during the first year in Work Package 2 (WP2) "*Enhancing Citizen science tools and methodologies*", and a detailed description of the design and early prototype of the CSSK which is designed, developed and maintained as a result of the work package.
In this section, we start by providing an overview of the objectives and tasks of WP2; then, we give an overall summary of the CSSK and the aimed usage of each tool in the kit; later, we describe the interaction of WP2 with the other work packages of the project. Finally, we provide a detailed description of the structure of the rest of the document.

1.1. WP2 overview

CSIC is the Lead Beneficiary of WP2, with Dr. Jesús Cerquides Bueno as its Work Package Leader. POLIMI, UNIGE and UNITAR are also contributors for this work package.

1.1.1. WP2 objectives and tasks

WP2 aims to fulfill the specific objective 1.1 in the Grant Agreement (GA), namely *"Enhance existing open source CS tools with artificial intelligence to analyze social media and other non-conventional big data sources, in particular for monitoring the impacts of extreme climate events, to be able to effectively process very large quantities of posts (millions) when the events are still in progress."*

As a result of the analysis of the most relevant needs for CS practitioners as well as the capabilities of the contributors, the GA also describes seven fine-grained objectives of WP2:

- *SO1.1a: develop software support for structured deliberation;*
- *SO1.1b: set up human-machine collaborative learning strategies to reduce the amount of effort requested from humans;*
- *SO1.1c: evaluate and improve the quality of data provided by CS projects;*
- *SO1.1d: combine existing computational techniques for social media data extraction for monitoring SDGs;*
- *SO1.1e: reduce the knowhow and resources needed by citizens wanting to start in grassroots initiatives to create new CS projects;*
- *SO1.1f: provide a Citizen Science Solution Kit using advanced AI techniques;*
- *SO1.1g: explore social media flows between citizens doing CS projects and AI components.*

The fulfilment of these objectives is accomplished in the GA through work in five different tasks, namely:

- *T2.1: Deliberation technologies for citizen science (CSIC, UNIGE, POLIMI)*
- *T2.2: Human-machine collaborative learning. (CSIC, UNIGE, POLIMI)*
- *T2.3: Agreement and data quality analysis (POLIMI, CSIC, UNIGE, UNITAR)*
- *T2.4: Self-composition. Adaptive services (POLIMI, CSIC, UNIGE)*
- *T2.5: Enriching Social Media content by Citizen scientist (POLIMI, UNIGE, CSIC)*

Tasks T2.1 and T2.2 are led by CSIC whilst tasks T2.3, T2.4 and T2.5 are led by POLIMI. A major result of the work in these tasks is the CSSK, which we describe next.

1.2. The Crowd4SDG Citizen Science Solution Kit

A *CS project leader* is a person, group of persons, or organization who is interested in carrying forward a CS project. The CSSK is a set of tools curated, and maintained by the Crowd4SDG project with the objective to help a CS project leader to successfully take forward a CS project, thus fulfilling objectives SO1.1e, and SO1.1f. Each of the tools focuses on a common recurring need of CS project leaders according to the experience in CS among the members in the consortium. Table 1.2 connects the CS project leader needs currently covered by each of the CSSK tools and shows its connection to the WP tasks and the WP objectives to which the work in each tool contributes. Notice that with the current set of tools under development we are aligning our work with each of the WP objectives, leaving no objective uncovered.

We also include two more tools in Table 1.2 which are not addressed by WP2 but form a part of the CSSK, namely CS Logger and Collaborative Sonar (CoSo). CS Logger is an open source data collection platform that makes it easy for anyone (without prior programming or design experience) to build and configure customized mobile applications for their CS projects. These applications can feature common "data collection" functionalities such as taking geo-located images and gathering additional information based on survey questions. CoSo, on the other hand, is a smartphone application aimed at understanding how team interactions impact team performance and learning. It looks for the answers to like how team members collaborate and how subgroups are formed. In addition, CoSo provides surveys to collect answers about qualitative team features such as diversity (demographic, skills) or organization (roles, relationships).

CS project leader needs to	Tool	Task	Objective
Increase the involvement of citizens into the management of the project	Decidim4CS	T2.1	SO1.1a
Enroll a volunteer community to perform a complex data classification task	Project Builder	T2.2	SO1.1b
Analyze data obtained from a volunteer community	Crowdnalysis	T2.3	SO1.1c
Extract visual evidence about a situation from images on Twitter, filtering out irrelevant images, and geotagging them	VisualCit	T2.4, T2.5	SO1.1d, SO1.1g
Document the progress of its CS project	Decidim4CS, SDG in Progress		
Create customized mobile apps for data-gathering CS projects without coding	CS Logger		
Explore how team interactions impact team performance and learning via a mobile app	Collaborative Sonar		

Table 1.2 Needs covered by the CSSK. Connection to tasks and objectives.

1.2.1. A CSSK case story

The following story represents what a typical story of the usage of the CSSK toolkit could look like:

Jane Doe wants to start a Citizen Science project to help her community in Ruritania be better prepared in case a flooding occurs as a result of climate change. In order to understand what the needs of the community are, she uses Decidim4CS, where she can easily set up a homepage and blog for the project, receive the proposals from the community, organize meetings and request citizens to vote. After three months of debating and self-organizing through Decidim4CS, it has been agreed with the community that better data is needed to make adequate decisions. In particular, they have co-decided that a map of the places which are more likely to get damaged when floods occur would be a very valuable asset for the project, and they are willing to work together to make it. They agree to build the map based on information of the last two floods that took place in Ruritania. By means of VisualCit they crawl the tweets containing images from those two floods, filter those images which do not contain images of the floods, and geolocate the tweets. After that, they are left with several thousand images. They create a project in Project Builder, upload the images and request a set of volunteers to connect to Project Builder and help them classify the images by labeling them with one out of five different levels of damage. They decide that, to improve the quality of the data, each image will be labeled by five different volunteers. Once the volunteers have labeled all the images, they use Crowdnalysis to provide them with advanced AI models to go from labels provided by each of the volunteers to a consensus opinion which takes into account the accuracy of the different volunteers, or specific characteristics of the images. After a consensus labeling is established for each image, VisualCit helps visualize a map coloring the different regions of Ruritania with different intensities based on their likelihood to get damage in a flood and visualizing the associated images from the last two floods. Based on the map, Jane decides….

1.3. Work Package 2 management

To make sure the work package reaches its objectives, we hold weekly work package meetings to coordinate the activities of the different partners and the research and development efforts.
Since the work package involves heavy technological development, one of the main management tasks in this first period has been establishing the shared development and deployment infrastructure and establishing the processes so that the partners of the project are able to conduct joint development. A summary of this is provided in Section 2.2.

1.4. Structure of the document

Next, Section 2 describes the design and early prototype of the CSSK. After that, Section 3 describes the case studies that have been used to test drive the different tools in the CSSK. After that, sections 4 to 8 detail the degree of advancement of the work in each of the tasks of the project. Then, Section 9 connects the work in WP2 with the remaining work packages of the project. Finally, we draw some conclusions in Section 10.

2. Design and early prototypes of citizen science tools

In this section we provide a detailed overview of the current state of the Citizen Science Solution Kit. We start by summarizing the evolution and the current state of the CSSK tools during this reporting period. Then, in Section 2.2 we detail the management of the collaborative software development applied to CSSK as a whole. Sections 2.3-2.7 provide the analysis requirements, design details and prototypes of each tool in the CSSK.

2.1. Summary of the evolution of the Citizen Science Solution Kit

Along the first 12 months of the project we have:

- Performed an analysis of requirements for the CSSK, taking into account which of the tools were already operational, the needs of the Crowd4SDG participants in the GEAR cycle and our commitments in the GA;
- Put in place the infrastructure, management and technological processes required to design, implement, extend and maintain the CSSK.
- Moved a previously non existent tool, Decidim4CS through design, implementation and deployment. Decidim4CS is currently in operational status and has 34 registered users at the time of writing this document;
- Moved two non-existent tools, Crowdnalysis and VisualCit, through design and implementation. Both tools are currently usable as prototypes;
- Maintained and added minor functionalities to two tools which were already operational at the start of the project: Project Builder and SDG in Progress;
- Promoted, advertised and communicated the CSSK in the following venues:
    - VisualCit presented in Tutorial at SocInfo 2020 (Polimi-UNIGE), October 2020;
    - Decidim4CS presented at the 3rd Annual Citizen Engagement and Deliberative Democracy Festival (watch video), December 2020;
    - Project Builder, VisualCit and Decidim4CS were presented online at the Geneva Tsinghua Initiative Master Workshop (gt-initiative.org), March 2021;
- Prepared future promotion and communication activities:
    - Decidim4CS to be presented at the CitSciVirtual 2021, May 2021;
    - VisualCit to be presented at the 43rd International Conference on Software Engineering, Track Sw Eng in Society, May 2021;
    - Decidim4CS to be presented and used to gather CS feedbacks for the Genigma project at the Barcelona International Youth Science Challenge, July 2021;
- Tested and evaluated the tools in two different case studies: the VisualCit early prototype for COVID-19 and the Albania Earthquake, described in Sections 3.1 and 3.2;
- Drawn valuable lessons for the case studies which result in new requirements. These lessons learned are reported in Section 3.3.

The evolution of the CSSK in the first 12 months is summarized in Table 2.1.1.:

Tool	Previous status	Current status	Main advances
Project Builder	Operational	Operational	Maintenance and minor functionality addition
Crowdnalysis	Non existent	Prototype	Design & Implementation
VisualCit	Non existent	Prototype	Design & Implementation
Decidim4CS	Non existent	Operational	Design, Implementation & Deployment
SDG in Progress	Operational	Operational	Maintenance and minor functionality addition
CS Logger	Non existent	Under development	
Collaborative Sonar	Operational for in-person events, Not operational for online events	Operational for in-person and online events.	Functionality addition

Table 2.1.1. Evolution of the tools included in the Crowd4SDG CSSK.

2.2. Management and development processes

Developing and maintaining the CSSK requires the coordination of a large team of scientists, software designers, developers (both UX and back-end) among other roles. In this section we overview the technological infrastructure put in place to manage the design, development and deployment of the different tools developed in this work package.

2.2.1. Development Process and Development Services

The development of the tools is managed in GitHub collaborative software development platform. We have created a GitHub organization Crowd4SDG (github.com/Crowd4SDG) which coordinates the source code for most of the projects related to the project. In order to save the costs of GitHub Large File Storage (LFS) service, we have moved the development of those projects requiring LFS to a GitLab server running in CSIC premises in Bellaterra (https://gitlab.iiia.csic.es/crowd4sdg). GitLab provides similar source code management functionality as GitHub, but it is an open-source software that can be installed locally. Any Crowd4SDG contributor is required to have an account either at GitHub or at the CSIC GitLab. Currently there are 16 contributors registered at the Crowd4SDG GitHub organization and 9 at the CSIC Crowd4SDG GitLab organization.

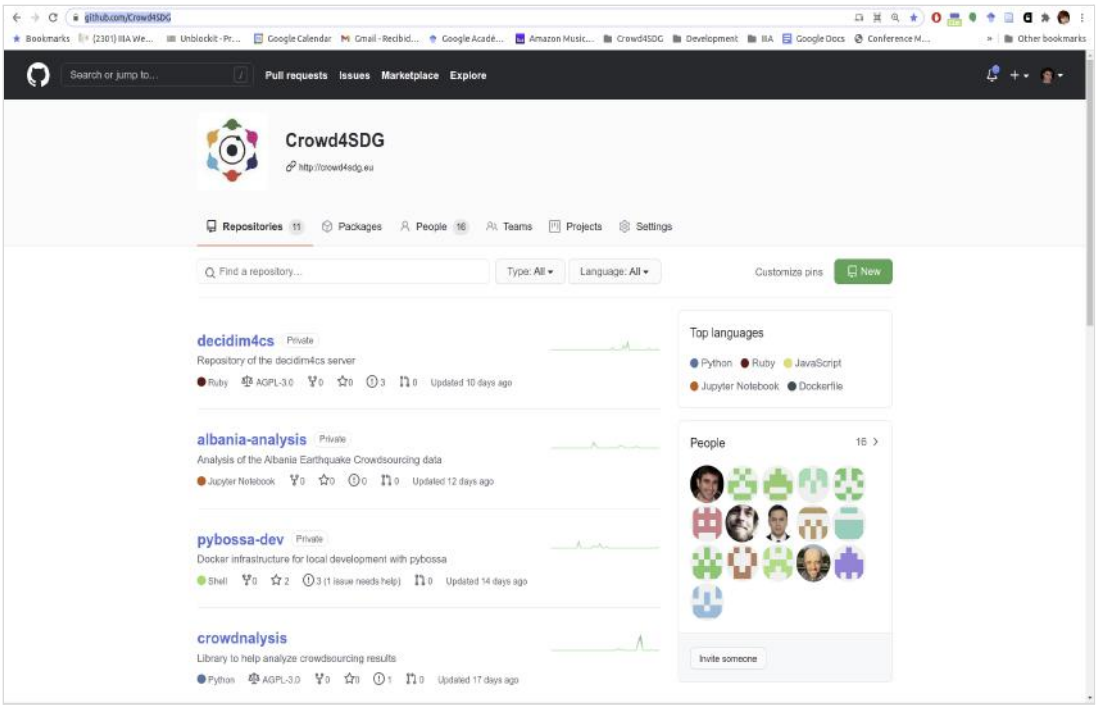

Figure 2.2.1. The Crowd4SDG GitHub organization

The branching model used in our repositories is the well known git-flow branching model.

2.2.2. Development and deployment infrastructure

To ease the management of the technological infrastructure, software deployment is automated and based on *docker* and *docker-compose*. Docker (https://docker.com) is an OS-level virtualization platform to deliver software in packages called *containers*. A container is a running instance of a Docker *image* which is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries and settings. Containers isolate software from its environment and ensure that the containerized software will always run the same, regardless of the infrastructure. *Docker-compose* is a tool for defining and running multi-container Docker applications.

For each running facility we have:

- A GitHub project hosting the code.
- Detailed instructions to bring up a development server in any of the developers machines by means of docker-compose.
- Detailed instructions to bring up a production server in a preproduction and production environment by means of docker-compose.

2.3. Project Builder

The Citizen Science Project Builder (PB) is a web-based tool that allows researchers, students, and all members of the public to create and run data-analysis Citizen Science Project. Such projects may take many different forms, from classifying images of snakes to transcribing handwritten German dialect, from collecting samples of water to taking pictures of insects and plants. Typically, volunteer contributors are asked to perform complex data

classification task (ie. classify, tag, describe, or geo-localize) that are still best performed by human minds and skills. In particular, the PB supports projects based on existing digital data that can be in the form of images, text, PDF documents, social media posts (tweets), sounds and video clips.
The PB provides an interface that requires limited technical knowledge, and ideally little or no coding skills. Its aim is to facilitate the co-creation of Citizen Science projects between the two communities of academic researchers and volunteer contributors, by starting with the implementation of simple pilots. Any idea for a data classification project can be implemented with little effort, the basic requirement being that the project responds to some simple criteria available in the platform. By building around it an initial community of contributors (colleagues, friends, family, and more!) the research question, process, data quality can be tested and iterated.
The PB Implementation is based on the open source crowdsourcing framework PyBossa and its code is publicly available under the 'CitizenScienceCenter' organisation on GitHub.
Since May 2020 the PB has evolved on several fronts, including bugs fixing, new functionalities, and enhancements to the user experience. Among others, these include additional flexibility for project owners (sharable link), multiple language support (non Latin alphabets and emoticons), simpler project browsing and full GDPR compliance. To support community's engagement, a Forum was also developed and activated for a few months to test the use and the effect of discussions on contributions and engagement.

2.4. Crowdnalysis

As explained above, the Project Builder facilitates the management, creation and run of Citizen Science projects that allow a volunteer community to perform complex data classification tasks. The main aim of Crowdnalysis is to ease the statistical analysis of the data gathered by the Project Builder, in order to help the scientist managing the project get the most accurate information from the contributions of volunteers. For example, imagine we need to annotate one thousand images from a natural disaster as either showing "no damage", "moderate damage" or "severe damage", and that we have a community of volunteers (workers[1]) that offer us their help. Each image is sent to five different workers, and each worker annotates it independently. The usual way to solve this problem is to use majority voting where each worker annotation is considered equally valuable. Thus, the consensus for an image is computed as the most common annotation. However, it is oftentimes the case that some workers are better than others at annotating images. The use of consensus algorithms different from the vote of the majority can turn out very profitable for CS projects. **The Crowdnalysis library is an open source software library to ease the use of advanced probabilistic consensus models in CS projects.**

2.4.1. Crowdnalysis analysis of requirements

The Crowdnalysis analysis of requirements follows a use case approach [Gomaa2011] for the functional requirements. The main actor in the Crowdnalysis analysis of requirements is a citizen scientist managing a Project Builder project. We will refer to him as *citsci*.

There are several use cases in which the Crowdnalysis library will play a role:

- UC2.4.1.1. A citsci wants to get the results of majority voting or other higher quality consensus algorithms for its project with minimum work.

[1] A worker is any of the participants in the annotation process. See section 5.1.4

- UC2.4.1.2. A citsci managing a large project with high data quality demands wants to use prospective analysis to decide on the most convenient project configuration.
- UC2.4.1.3. A citsci managing a large project with high data quality demands wants to use intelligent scheduling to decide intelligently which task to assign to an incoming worker.

Also a set of non-functional requirements have been identified such as:

- NFR2.4.1.1. Crowdnalysis should easily integrate with Project Builder.
- NFR2.4.1.2. Crowdnalysis should be able to deal with Project Builder projects with up to 100K tasks.
- NFR2.4.1.3. Crowdnalysis should be open source software.
- NFR2.4.1.4. Crowdnalysis should be easy to use and maintain.
- NFR2.4.1.5. Crowdnalysis should be extensible with additional models.

2.4.2. Crowdnalysis design

Based on the objective of satisfying the functional requirements in the use cases presented above as well as the non-functional requirements, we have designed Crowdnalysis to be a software library implemented in the Python programming language (this provides ease of use and maintenance, since it is a widely known programming language, and it eases the integration with Pybossa which is also developed in Python).

In order to extend Crowdnalysis with additional models (NFR2.4.1.5), we have abstracted the interactions by means of the `AbstractConsensus` class (see Figure 2.4.1.1).

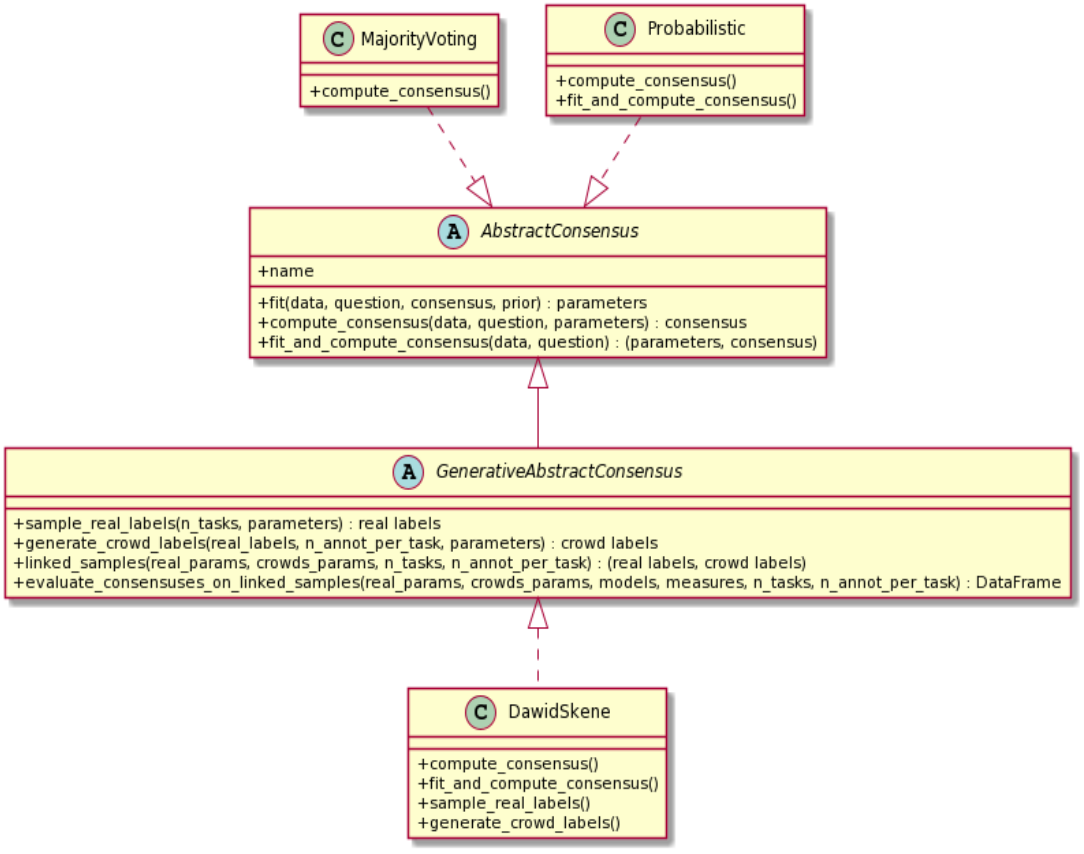

Figure 2.4.2.1. Class diagram of the Crowdnalysis library

From a mathematical standpoint, Crowdnalysis `AbstractConsensus` class provides a computational representation of the probabilistic model presented in the journal paper that has been published in the high quality (JCR first quartile) Mathematics journal [Cerquides2021]. And depicted below in Figure 2.4.1.2 by means of a probabilistic graphical model.

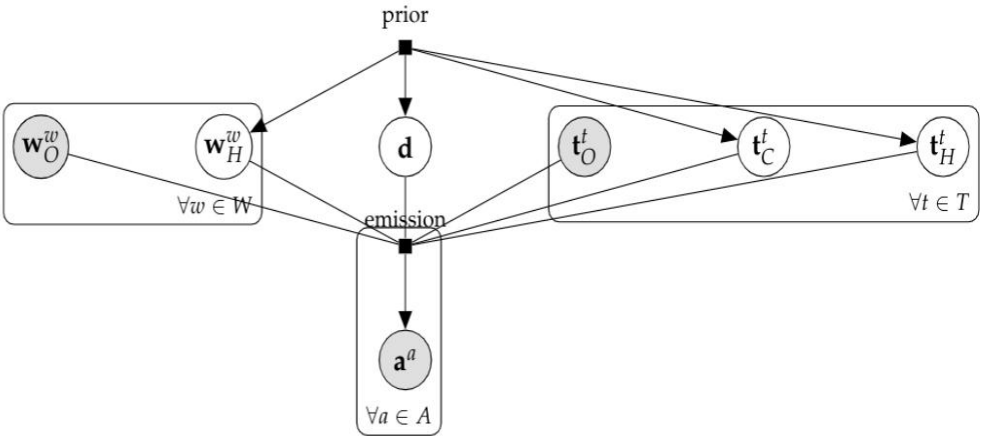

Figure 2.4.1.2. Probabilistic graphical model description of the abstract consensus model.

On the other hand, `GenerativeAbstractConsensus` subclass in Figure 2.4.2.1 is created to automate prospective analysis that satisfies the requirements of UC2.4.1.2.

2.4.3. Crowdnalysis prototype

Crowdnalysis is implemented as a software library and currently it is available for download and use from https://github.com/Crowd4SDG/Crowdnalysis. It has been used in the Albania earthquake use case described in more detail in Section 3.2.

2.5. VisualCit

The VisualCit (Visual Citizen) is a framework to extract images from social media and filter and classify them combining AI tools and crowdsourcing. It is based on a data analysis pipeline that retrieves information from social media, preprocess it to make it ready for crowdsourcing, and finally aggregates the results and provides visualization on maps and tools for validation.

2.5.1. Requirements

The first year of the project was devoted to design and test the VisualCit basic pipeline to be made available in Gear 2.
Starting from the previous experience of participants in the project, developed within the E2mC project and two Hackatons in which UNIGE and Polimi participated in preparation for the project in April the initial requirements were collected:

- Requirement 1. Need to define a common format for data exchange among tools. Alternatives are data APIs, csv files, management of different character sets
- Requirement 2. Images safe for work

- Requirement 3. Provide citizen science with an adequate number of images/posts to be analyzed. The number must be sufficient for the analysis, but not too many for the crowd.
- Requirement 4. Flexible combination of tools
- Requirement 5. Distribute crawlers to users. Users must have their own credentials to access social media
- Requirement 6. Need for methods to assess the results of the pipeline
- Requirement 7. Boost crowdsourcing capabilities: considered alternatives are creating communities, using existing ones (paid/not paid), creating automatic classifiers from initial images classified by the crowd using them as training data. In addition providing more agile annotations tools to the crowd is a need.
- Requirement 8. Interactive interfaces:initially considered for crawler and simple text-based interfaces for input and output pipelines

2.5.2. VisualCit Design

VisualCit has been designed as a set of separate components, exchanging data in a predefined format.

The general structure of VisualCit is illustrated in Fig. 2.5.1. As shown in the figure, the developed tool is composed of 11 components of the types described in Task 2.4.

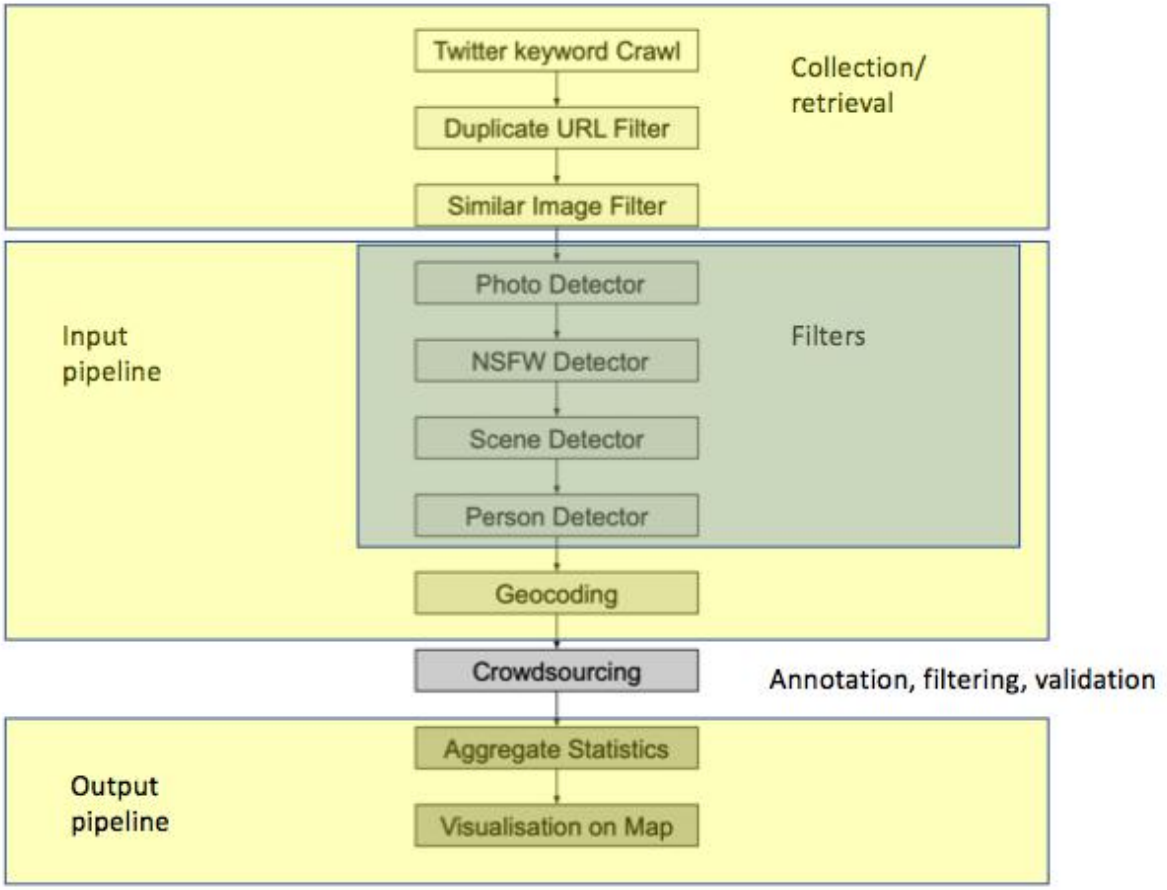

Fig. 2.5.2. VisualCit prototype pipeline

A data Interchange format has been defined (see [Pernici2021] on Zenodo for details), using csv files.

Each execution of the pipeline on a dataset is performed on an input dataset, splitting the data in batches of 50,000 posts

The components orchestration is based on a service-based approach.
The Twitter crawler and the geocoding components are available as web services, with a set APIs defined for them, as well as interactive web-based interfaces.
The ML classifiers are provided as docker components. The service composition is performed through Python scripts in the first prototype.

Crowdsourcing is performed using PyBossa using the Project Builder.
Visualization and annotation can also be performed with an interactive web-based service developed in the project.

The Output pipeline: merges PyBossa results (with a majority voting strategy to aggregate crowdsourcing results) and creates thematic maps.

2.5.3. VisualCit Prototype

The social media pipeline developed in Year 1 has been created in view of its applications in Gear 2 and Gear 3 of the project, developing a set of components that can be combined in a flexible way and can be used as services.
The developed framework includes the modules illustrated in Task 2.4 and Task 2.5 descriptions and has been tested in two case studies illustrated in Section 8 (Covid-19 and Albania case studies).

The code is available in the following repository for the components developed in the project: https://gitlab.iiia.csic.es/crowd4sdg/polimipipeline

A detailed description of the VisualCit pipeline and its components is reported at https://docs.google.com/document/d/1HJ_xgJl-7GdDurenSHOo72wcuyfJHW9706TLP9X7wLI/edit?usp=sharing and available as documentation in GitLab.

Two components (Twitter crawler, geocoding) are provided from background material available for the project and have been provided as services integrated and adapted within the project, providing outputs adapted to be compatible for integration with the other components of the pipeline:

Crowd4SDG Twitter crawler is available as a service at:
http://131.175.120.2:20002/static/index.html. It is provided as a webservice with APIs and as a software component developed in Python for deployment on machines owned by Citizen Scientists. Given Twitter regulations each user must use the tools with their own Twitter credentials.

An *image visualizer/annotator* is provided as an interactive service (generic one, multiple images) https://social-distancing-project.herokuapp.com/

The *PyBossa* version available as a service for the project is run by the Citizen Science lab of Zurich, using and adapting the Project Builder interface. Several projects were created for the COVID-19 case study for different weeks of analysis. An example of the created projects is the following: https://lab.citizenscience.ch/en/project/60

A server has been set for the project at PoliMi to facilitate experimentation with the available tools. An instance of PyBossa has been created there for experimentation with new functionalities.

All the ML tools for *filters* described above require a Docker environment that is supported by the server. The models are available through the GitLab repository indicated above.

The input pipeline and the output pipeline have been realized as separate Python scripts invoking the components. For each, it is possible to start and end at any point of the pipelines described in the figure, selecting the options with a text-based interface.

Accounts for setting up new experiments in the GEAR cycles can be requested by writing to crowd4sdg@polimi.it
Further information for citizen scientists and thesis students are available at: http://pernici.faculty.polimi.it/crowd4sdgpolimi/

2.6. Decidim4CS

Decidim4CS is a digital platform for participatory citizen science. Decidim4CS constitutes the deliberation technology used in Crowd4SDG, and it relates to the Task 2.1 (Deliberation technologies for citizen science) of the WP2. Specifically, it allows citizen scientists to organize themselves democratically by proposing and discussing ideas, scheduling meetings, conducting surveys, making decisions through different forms of voting, and monitoring the implementations of these decisions.

2.6.1. Decidim4CS analysis of requirements

The main actor in the Decidim4CS analysis of requirements is a citizen scientist who wants to organize a new citizen science project or participate in an ongoing project. We will refer to him/her as *citsci* as we did for Crowdnalysis before.

We can highlight the use cases for Decidim4CS as below:

- UC2.6.1.1. A citsci has an idea that he/she believes could make an impact for a better world, and wants to hear public opinion on his/her ideas.
- UC2.6.1.2. A citsci wants to contribute to ongoing citizen science projects that matter most to him/her.
- UC2.6.1.3. A citsci wants to monitor the progress on the project goals he/she created or contributed.

Several non-functional requirements can also be listed:

- NFR2.6.1.1. All shared ideas, votes for ideas should be public.
- NFR2.6.1.2. Moderation should be possible for spam content.
- NFR2.6.1.3. Decidim4CS should be open source software.
- NFR2.6.1.4. Decidim4CS should be user friendly to permit maximum public engagement.
- NFR2.6.1.5. Decidim4CS should be customizable for the project or institution needs.

2.6.2. Decidim4CS design

Decidim4CS is based on decidim (https://decidim.org) which is a free open-source software originally created by the Barcelona City Hall as a participatory democracy platform for cities and organizations (NFR2.6.1.1). It is designed to be used primarily by city councils to resort to public opinion or vote for the city management activities. Besides, in order to be used by different institutions, it is designed as a highly customizable platform.

Decidim's back-end allows a plethora of options to customize the application to meet the organization's needs. The configuration possibilities include adding/removing application's core components (e.g. Projects, Proposals, Surveys, Meetings), renaming these components, modifying the look-and-feel of the user interface, language localization, user management, data exporting, etc.

When there is a further need to modify the application's behaviour, this can be done by a software technique called "monkey patching". This technique allows to modify the behaviour of a software at runtime without changing the original source code. The Ruby programming language in which decidim is written permits monkey patching.

Decidim4CS leverages decidim's configurable and extendable nature and adapts it for use in citizen science (NFR2.6.1.4, NFR2.6.1.5). For the sake of compatibility with future releases of decidim, Decidim4CS is customized largely by back-end configuration, and monkey patching is applied only when necessary.

In particular, we have deactivated certain decidim components and renamed some others so that they suit better for citizen science (NFR4, NFR5). We also fit the terminology of the internal messages to our needs. Moreover, we created demo projects where we used all relevant components like proposals, debates, surveys, meetings, etc. We will give examples to these components in Section 2.6.3.

2.6.3. Decidim4CS prototype

Decidim4CS hosts citizen science *projects* (UC2.6.1.1, UC2.6.1.2). Each citizen science project can consist of whole or a subset of the following components:

- *Informative pages*: e.g. Describe the overall project;
- *Meetings*: Schedule and share meetings;
- *Proposals*: Propose methods for a project to achieve its goal;
- *Debates*: Discussions on specific issues, questions, ideas;
- *Blog*: Create chronological news items regarding a project;
- *Accountability*: Monitor the progress of a project or its sub-goals (UC2.6.1.3);
- *Survey*: Configure and conduct online surveys.

Decidim4CS also provides the project participants with the following mechanisms of participation that are available to the above components where relevant (UC2.6.1.1, UC2.6.1.2):

- *Support/Endorse*: Demonstrate a positive agreement in accordance with the proposal. These can be counted as votes.
- *Comment*: Share your idea on a proposal, debate, result, meeting etc. with the options of being in favour, against, or neutral. Comments encourage deliberation. Each comment can be further remarked as "agree/disagree" by other participants.

- *Share*: Share the component on major social media platforms or any other by acquiring its direct link.
- *Follow*: Get notified upon updates of proposals, debates, meetings, blogs

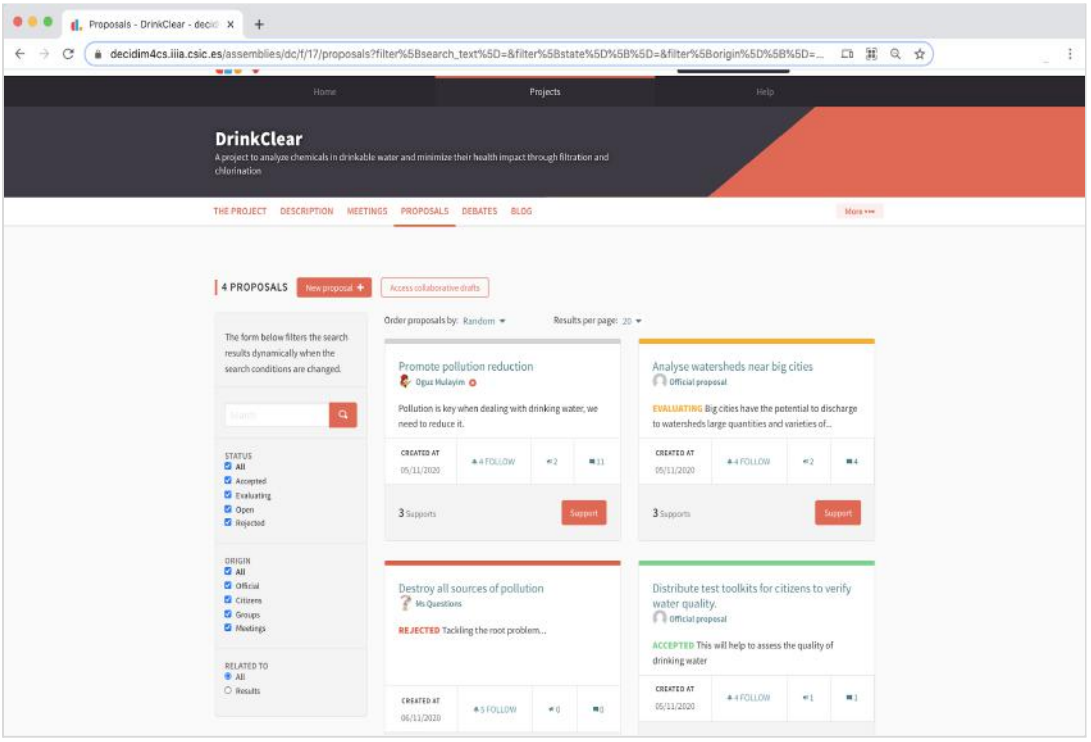

Figure 2.6.3.1. Proposals at four possible states: Open, Evaluating, Accepted, and Rejected

Proposals are the core component of Decidim4CS. They can be created by the administrators or registered participants. Once the deliberative process starts on an *open* proposal, it passes to the *evaluating* state. After deliberation by public debates, comments, supports, etc., the proposal is either *accepted* or *rejected*. See Figure 2.6.3.1 for the proposals in the demo project. Figure 2.6.3.2 shows the supports for a proposal and the ongoing debate, while Figure 2.6.3.3 gives an example of a scheduled meeting. Meetings can be scheduled ad-hoc or periodically. Moreover, Decidim4CS also allows the moderation of the comments (NFR2.6.1.2).

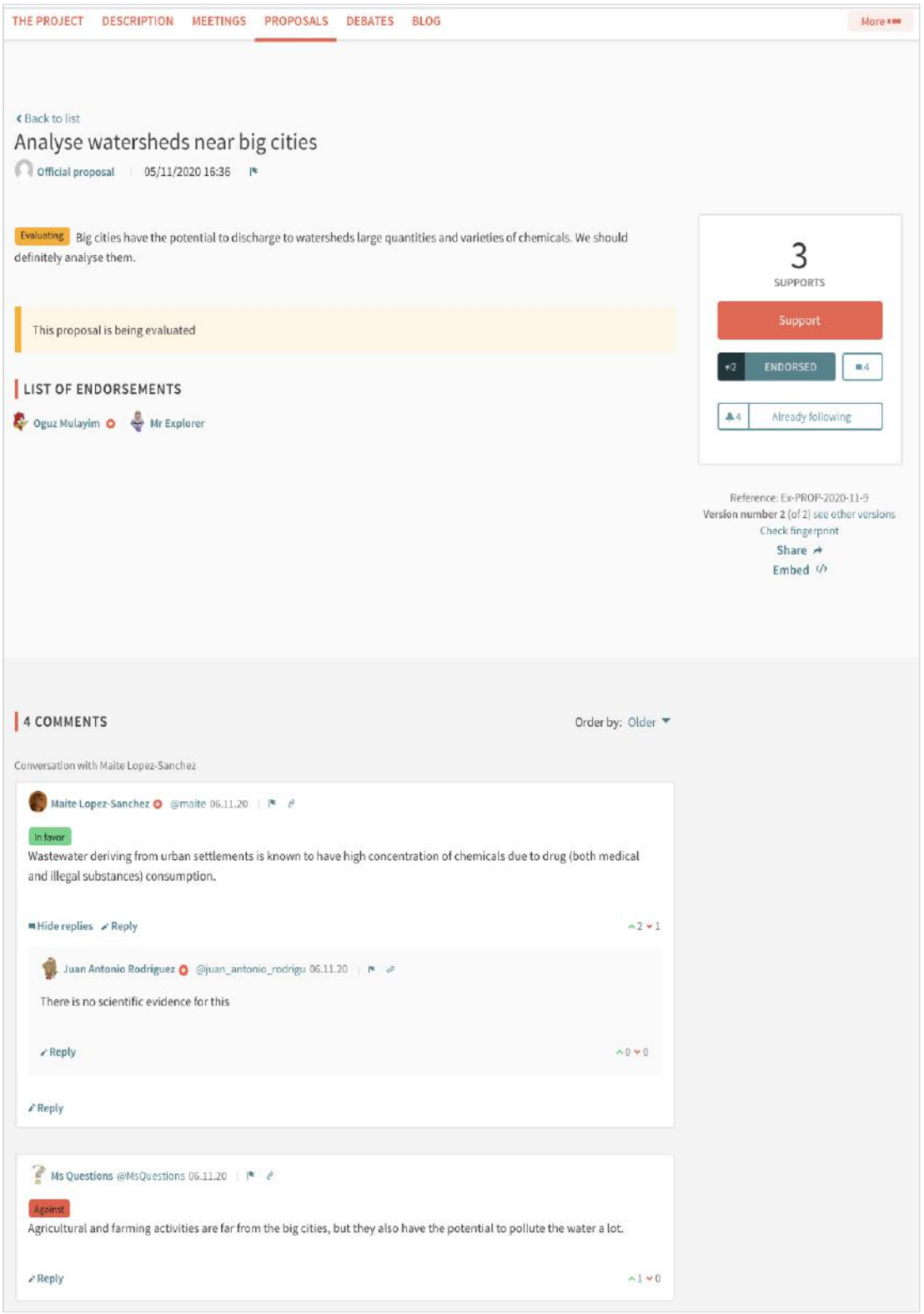

Figure 2.6.3.2. Supporting a proposal and commenting on it.

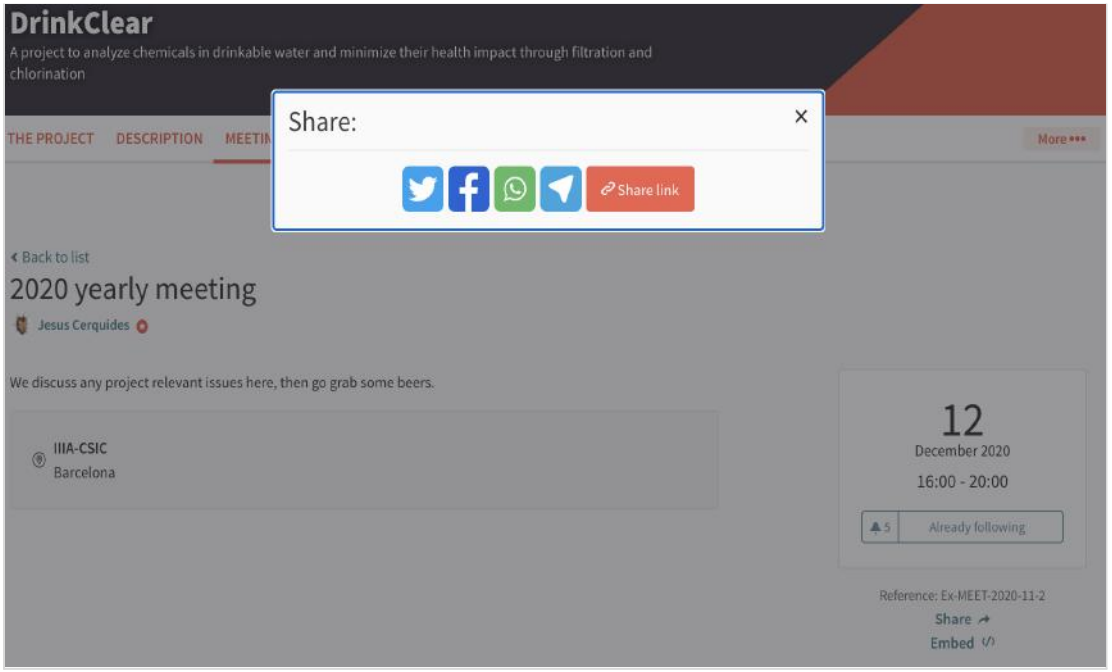

Figure 2.6.3.3. Scheduling a project meeting and sharing it on other media.

Decidim4CS runs on a virtual machine at the IIIA-CSIC's Barcelona premises. Decidim4CS application code, docker images and docker configuration files reside at the public GitHub repository https://github.com/Crowd4SDG/Decidim4CS (NFR2.6.1.3). The repository is kept up-to-date at all times.

2.7. SDG in Progress

SDG in Progress is a platform for anyone who wants to document projects they are doing towards tackling the UN Sustainable Development Goals (SDGs), or who wants to get inspired by other people's projects, re-use them or re-purpose them.

The platform is based on Build in Progress, which was developed for documenting maker projects by Tiffany Tseng of MIT Media Lab's Lifelong Kindergarten, as part of her Ph.D. thesis. It has been re-purposed by Oday Darwich of University of Geneva's Citizen Cyberlab, as part of the Geneva Tsinghua Initiative (GTI) for the Sustainable Development Goals.

There are many ways to document projects: wikis, GitHub etc. The point of SDG in Progress is to build up an open repository of SDG projects, which may involve hardware and software development, or focus more on grassroots or policy initiatives. The main idea is to inspire creativity and promote co-creation.

3. Case studies

This section describes the two case studies where our toolset has been used in Sections 3.1 and 3.2 and draws lessons learned from them in Section 3.3.

3.1. VisualCit early prototype for COVID-19 case study

Given the ongoing COVID-19 outbreak, it is essential for governments to have access to reliable data on policy-adherence with regards to mask wearing, social distancing, and other hard-to-measure quantities. In this paper we investigate whether it is possible to obtain such data by aggregating information from images posted to social media. The paper by Negri et al. [Negri2021] presents the developed VisualCit pipeline for image-based social sensing to discover in which countries, and to what extent, people are following COVID-19 related policy directives.
The obtained indicators resulting from VisualCit were compared with the indicators produced within the CovidDataHub behavior tracker initiative. Preliminary results show that social media images can produce reliable indicators for policy makers (see Fig. 3.1.1).

Fig. 3.1.1 General overview of the analysis in Covid-19 case study

In the case study we use a citizen science approach to complement the collection of information from Twitter asking the crowd to evaluate the behaviour of people in the extracted images. A series of questions is posed to the crowd workers, to assess the visible behaviour of the people. In particular we focused on social distancing and the use of face masks, as these data are difficult to extract automatically (see for example existing challenges on mask detection, e.g. https://www.aicrowd.com/challenges/mask-detection-challenge) and in many cases requires human judgement.

The open source PyBossa (https://pybossa.com/) platform for human data mining has been adopted in this project, with the extension of the Project Builder realised at the Citizen Science Center Zurich for an easier creation and management of crowdsourcing projects. Each post is shown to the crowd worker with the image and a proposed geolocation for it (see figure). A series of questions concerning the image contents related to the Covid-19 pandemic are proposed to the crowd worker, concerning social distance and face mask usage.

The main challenges posed by the case study drove the development of the VisualCit tool. In particular challenges were related to the size of the collected Tweets (arount 1.5 million a week) and the fact that many of the retrieved images were not relevant to assess social behavior. For the details of the study we refer to the published paper [Negri2021] derived from the research developed in Year 1 of Crowd4SDG. Here we focus on the VisualCit developed tool.

The VisualCit pipeline has the goal of providing input to crowdsourcing, with an input pipeline to filter posts, and collecting the results from crowdsourcing to build indicators and build thematic maps.

The validation analysis has been performed separately against the Covid-19 Behavior Tracker initiative (CovidDataHub project) by the Institute of Global Health Innovation (IGHI) at Imperial College London and YouGovSurveys to analyze the quality of the results.

The details are reported in [Negri2021] The dataset of three weeks analyzed in detail in the paper is available in Zenodo [Pernici2021]. As discussed in the paper, the VisualCit pipeline was able to extract mask usage indicators with a good correlation with indicators in the CovidDataHub surveys for the common countries, and to retrieve further information about countries not present in the CovidDataHub surveys. It demonstrates a first attempt to derive numerical indicators from visual evidence of a situation, which has a potential for application for SDGs where NSO have problems collecting data.

The three datasets analyzed in the study provide 14961 annotated posts by at least 3 persons. The total size of the crowd is 90 persons, counting anonymous annotators as a single contributor. Out of the crowdsourced posts, around ⅔ can be considered useful for statistical analysis after aggregation and verification of results.

The size of the three datasets and the results of the filtering is illustrated in Table 3.1.

Date	tweets	filtered-geoloc	reduction %	crowdsourced %
May 13, 2020	470,255	25,541	94%	13%
Aug. 1, 2020 (Phase 2)	NA	5,412	NA	100%
August 17-23, 2020 (w34)	1,465,494	49,749	96%	10%

Table 3.1 COVID-19 datasets

As shown in Table 3.1, given the size of the crowd it was possible to analyze only a fraction of the three datasets. In addition, other datasets for the same case study, with weeks ranging from 34 to 45 of year 2020 have been collected, in order to follow the evolution of the situation. Covid-19 datasets are assessed by WP5 in the deliverable D5.2 - *Data usability*

assessment and recommendations for SDGs for GEAR cycle 1, and deemed to meet the criteria to be used "by NSO as a non-official data source".
As the size of the crowd is deemed to remain a constraint in the evolution of the project, we started studying how to build classifiers automatically on the basis of crowdsourced information, for the questions posed in the case study

The first promising results of the automatic construction of classifiers to complement crowdsourced data are reported in [Scuratti2021a, Scuratti 2021b] and are shown in Fig. 3.1.2.

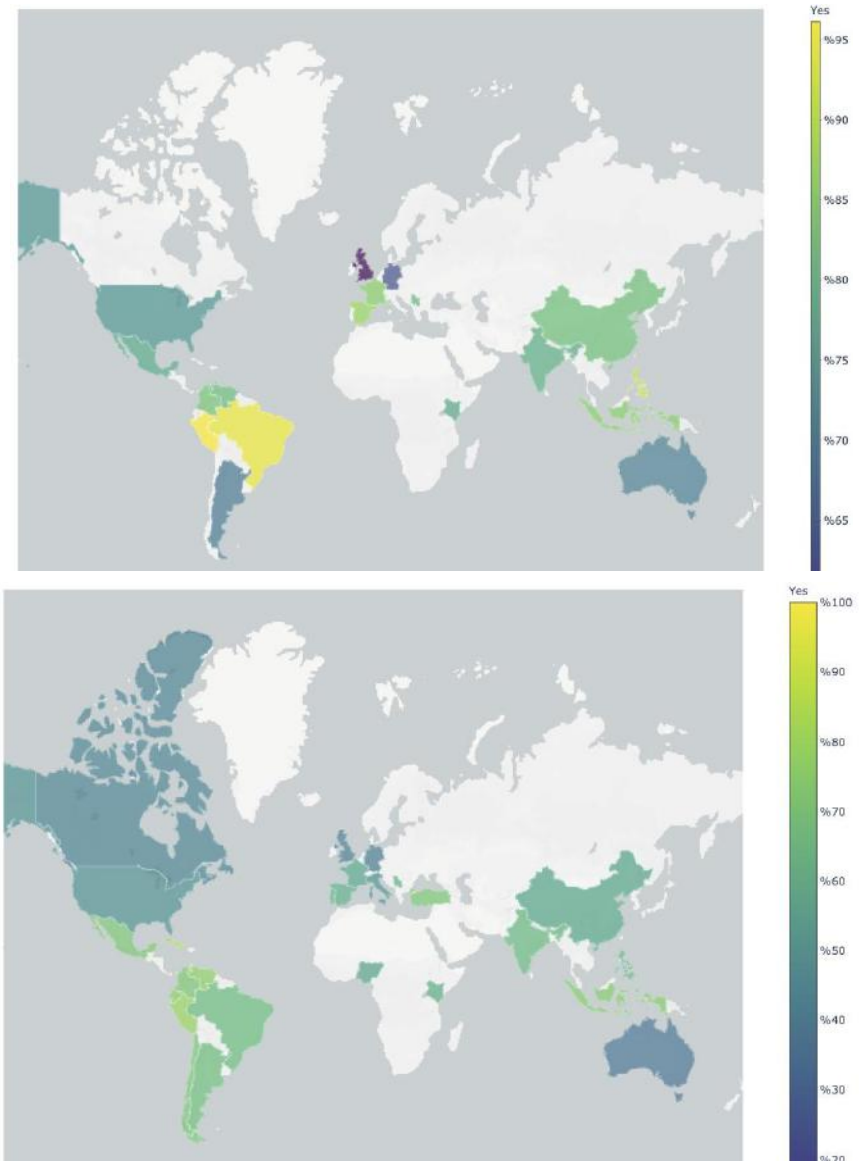

Fig. 3.1.2. Comparison of maps for week w34 (third week of August) of Year 2020 built with the VisualCit pipeline (top) and with adding a ML component after crowdsourcing (bottom).

The map on top in Fig. 3.1.2 is derived from VisualCit pipeline and crowdsourcing (threshold 20) and the one at the bottom adding an automatically constructed classifier from crowd annotations (threshold 250). The map represents Yes values, i.e., percentages of all persons wearing masks. A significant increase of the number of covered countries is shown. As

shown in [Scuratti2021a, Scuratti2021b] the dynamically added classifier allows improving the precision of the results when compared to CovidDataHub data.
The generated map shown above is an interactive map in which for each country the Yes and No indicators (percentages) can be retrieved.

3.2. Albania Earthquake case study

A second case study in which Crowd4SDG tools were applied is a scenario of extraction of images related to an emergency event from Twitter. The goal is to provide visual evidence derived from tweets on a natural disaster, namely an earthquake. This approach can be extended to other natural disasters related to climate change such as floods and storms. Visual evidence is useful for first responders and to provide information for rapid mapping operators [Havas2012, Zahra2020], and the derivation of indicators is envisioned (e.g., the number of severely damaged buildings, according to the Copernicus EMS classifications).

We have considered as a case study a dataset of 907 images derived from tweets related to the earthquake that struck Albania on 26 November 2019. The images were extracted from social media by the AIDR system of the Qatar University [Imran2014], and were already filtered both manually and automatically to evaluate their relevance to the event, and in particular for grading the damages shown in the images with an ML classifier as {Mild, Severe} [Imran2020].

In our case study, we incorporated the human-in-the-loop concept to further assess the images in the dataset. The images were given to three different crowds; namely *experts*, *volunteers* and *paid workers*, and they were asked to annotate the images assessing the severity of the damage seen on an image by using five different labels in {irrelevant, no-damage, minimal, moderate, severe}. For labelling the images, the experts and volunteers used the Crowd4EMS platform [RaviShankar2019], whereas paid workers were referred to the Amazon Mechanical Turk platform (MTurk).[2]

3.2.1. Harnessing Crowdnalysis

We applied Crowdnalysis to the labelling data to infer the following results:

1. A *ground truth* (consensus) on the image labels via experts' labelling data;
2. *Error-rates* of each crowd with respect to the ground truth;
3. A *prospective analysis* for the accuracy that we would expect from the three communities when they contribute with more annotations.

We note that the first and third points above are related to the use cases UC2.4.1.1 and UC2.4.1.2, respectively, in Section 2.4.1.

Figure 3.2.1.1 depicts the error-rates of the experts calculated by Crowdnalysis using the Dawid-Skene model [Dawid1979] that appears on Figure 2.4.2.1. We observe that although irrelevant and severe damage labels are more likely to be correctly annotated, the expected accuracies for other answers are below 50%. These results show that the *expert infallibility* assumption which is sometimes taken as granted in citizen science, is just a myth [Aroyo2015], and we can identify this in our case thanks to Crowdnalysis.

[2] Amazon Mechanical Turk: https://www.mturk.com/

Figure 3.2.1.1. Error-rates of the experts in labelling the severity of damage.

Figure 3.2.1.2 shows the results of the prospective analysis for different numbers of expert annotators. We see that Crowdnalysis yields higher quality of correct labels compared to the standard Majority Vote scheme for any number of annotators. We can say that with the probabilistic model provided by Crowdnalysis, we do not only achieve more accurate consensus results, but we also achieve them with less annotators. Therefore, using Crowdnalysis to infer consensus is more cost-effective than the standard majority voting scheme.

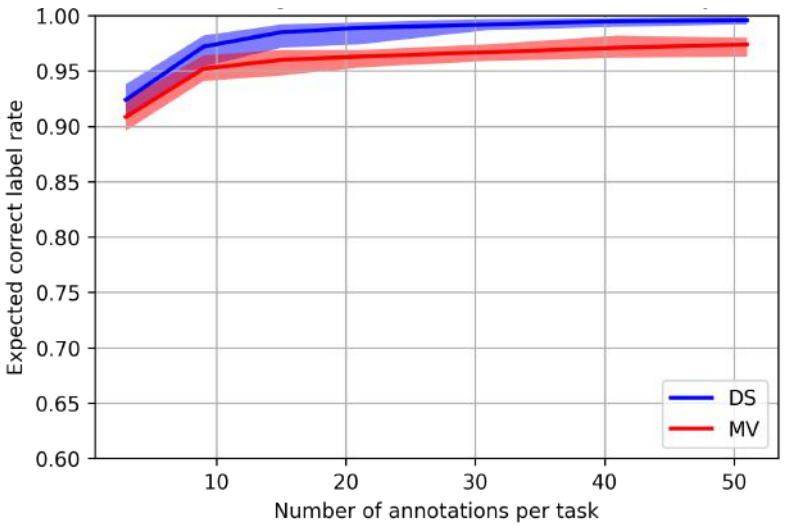

Figure 3.2.1.2. Prospective analysis for correct label rates for experts with Dawid-Skene Model (DS) and Majority Voting (MV) used for consensus.

A detailed description of the analysis and the interpretation of the results can be seen in the paper that has been published in the high quality (JCR first quartile) Mathematics journal [Cerquides2021]. We will soon publish the dataset used in the paper to Zenodo and make it publicly available. A detailed analysis of this dataset regarding its potential use by NSOs can be found in the deliverable D5.2.

3.2.2. VisualCit experimentation

A requirement of this case study is to allocate to the crowd tasks to be performed in an adequate number and quality of images to analyze: the capacity of the crowd to analyze images is limited by its size, and motivation is an important issue, i.e, tasks should be interesting enough to be performed. In fact, images from social media are often repetitive, or similar (e.g. snapshots of other posts), irrelevant (emoticons, memes, and the like). As a result, the problem in this type of scenario is to balance precision and recall in using classifiers, deciding which ones to apply, and deciding when to introduce crowdsourcing and which type of human evaluators to select.

We tested the VisualCit input pipeline in this domain, to evaluate its generality and the impact of its application. The goal is that of reducing the number of non relevant images with a limited loss in terms of recall.

The following ML filters were selected: Not Safe For Work (NSFW), Public, Photos. Deduplication was not selected as the initial dataset was already curated. No adaptation to the specific domain was made on the available filters. The textual interactive interface for selecting the initial and final steps of the input pipeline was used. The Qatar dataset was transformed into the VisualCit format with a simple script, some Tweet rehydration from tweet ids was needed.

Fig. 3.2.2. Relevance classification and pipeline application

The results are given in Fig. 3.2.2. The False labeled images are the ones that were found irrelevant according to the experts' consensus detailed in section 3.2.1., and True labeled ones are the relevant ones (i.e., labeled by one of the other four damage labels). In other words, experts found relevant only 605 of the 907 images that was deemed by the AIDR system. The figure shows that 654 images of the original 907 images are retained by VisualCit. In the figure we also see the impact of applying the pipeline has an impact on the precision of relevant images improved from 80% to 90%, while 64% of irrelevant images were discarded.

We also evaluated the impact of applying the geocoding component to the case study. It is known from the analysis in [Scalia2020] that geocoding has a positive impact of reducing non-relevant tweets and in identifying the images related to this area. In the present case study, as the dataset was already curated and the images were already referred to the area of interest, it is not very useful to further apply geocoding, as only in a limited number of cases the exact location could be detected. As a result 200 tweets out of the 600 relevant tweets were retained, with Albania as a location as a country and only in a few cases a more precise location identified. On the other hand, the experiment showed that the multilingual country

detection has been working well, as Albania was recognized also from posts mentioning the country name in Albanian (Shqipëria).

3.3. Distilling the case studies. Conclusions and new requirements

In the first year of the project, as the effort of WP2 was mainly devoted to develop tools to be used in the next GEAR cycles, the case studies illustrated above where derived either from crowdsourcing applied to an existing dataset (the Albanian case study) or to a new case study focuses on a current emergency situation (COVID-19 case study) in particular to devise techniques to extract indicators from social media images.
One of the critical aspects emerging from the application of the case studies was the difficulty of creating a crowdsourcing community to analyze a large number of posts. As this could be a critical aspect for the future, this will be considered both from the point of view of specific projects are they potential for involving communities, and from a technical point of view for developing new requirements in the next steps of the project.
The tools were also presented to potential users in a number of initiatives (see WP6 deliverables) and further requirements emerged from these presentations. In addition, research from WP5 considering interviews with NSO and other stakeholders are considered in the following considerations. For example, recently on May 27th, UNITAR hosted the panel "*Unleashing the potential of Citizen Science Data for monitoring the SDGs*" to discuss how to leverage citizen science data and the data quality assurance criteria. More than a hundred people from several NSOs and other organizations attended. The report on the CS data within the Crowd4SDG project was presented, and it received positive feedback from the attendees.
In the following, we discuss the new requirements concerning WP2 future work.

3.3.1. Crowdnalysis new requirements

The usage of Crowdnalysis in case studies has allowed us to identify three additional requirements which we highlight below.

1. A major issue that has been noticed is that the probabilistic models implemented by Crowdnalysis suffer from label switching [Stephens2000], as happens with many mixture models. Label switching leads to degradation in the quality of the predictions, and, at times, to extremely inaccurate estimation of the uncertainty. We have identified two ways to tackle this issue. On the one hand, we can apply general techniques, such as the ones implemented by the R package label.switching [Papastamoulis2016]. On the other hand, we can take advantage of the specific characteristics of our consensus problem in order to build probabilistic models which are not prone to label switching. We have decided to opt for the second one, and thus an additional requirement has been added to the tool, namely:

   UC3.3.1.1. A citsci should be able to select label-switching-free models for computing the consensus.

   We are already well ahead in the mathematical definition as well as in the implementation of label-switching-free models.

2. A second issue we identified is as follows. In many crowdsourcing projects, the workers or volunteers are given three options when a question is asked to them: 1) provide a proper answer to the question; 2) say that they do not know the answer; 3)

say that they do not have a clear answer. Most of the consensus models up-to-date assume that the set of real classes is exactly the set of labels available to the workers which also include the answer options 2 and 3. Thus, they model the error-rates for these two answers as well, and the result is prone to misinterpretation, if not useless. Since this is a commonly recurring theme in crowdsourcing, we have added a new requirement:

UC3.3.1.2. A citsci should be able to run analysis in which the real classes are different from the set of labels.

3. Finally, a third observation is that although the existing models implemented in Crowdnalysis undoubtedly provided added value, it is often the case that specific uses of Crowdnalysis will require the implementation of additional specific models (such as hierarchical models). This reinforces the non functional requirement NFR2.4.1.5. We are working on implementing this via a mapping to the Stan [Carpenter2017] modeling library.

3.3.2. VisualCit new requirements

The following requirements emerged from the first experimentation of VisualCit and its presentation in a tutorial and public events.

UC3.3.2.1: Dynamically configurable pipeline with a graphical user interface (GUI)

In the current version of VisualCit both the input and the output pipelines can be activated either in a programmatic way or with a simple text-based interface. The need emerged to be able to show the results emerging from the visual pipeline directly to the user, to facilitate its adaptive configurations giving a visual feedback on the results.
This requirement is combined with previous *Requirement 8. Interactive interfaces:* Interactive interfaces envisioned for the single components of the pipeline have to be combined with Year 2 - *UC3.3.2.1*

In Year 2 of the project, the work on the adaptive pipeline will focus also on providing a visual interface. An initial mockup for it is shown in Fig. 3.3.1:

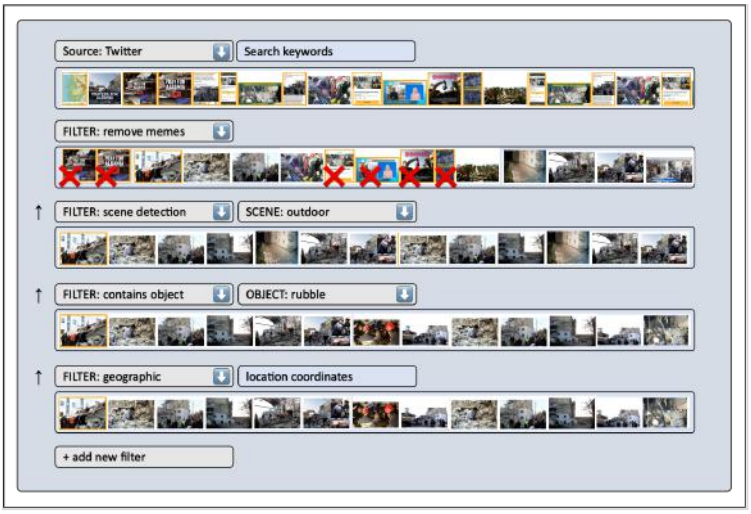

Fig. 3.3.1 Envisaged Graphical User Interface (GUI) for simple creation and reconfiguration of a social media pipeline.

UC3.3.2.2: Provide ad hoc services for GEAR projects
From GEAR 1 proposed projects, it is clear the each project will have specific needs in using the pipeline, for instance using different social media, datasets availability guidance, and so on.

In coordination with WP3, we envision to support the use and adaptation of existing services for specific project needs, contributing to refine their requirements and in developing mockups for data collection.

UC3.3.2.3: Focus on assessment of confidence
In WP5 the need emerges that not only indicators should be collected, but also the level of confidence of the results should be assessed, including possible biases due to the different coverage of social media in different countries.

UC3.3.2.4: assess the results of each tool in detail
To support the adaptive composition of tools, the results of each tool need to be assessed to assess their features: reduction rates for filters, quality of results, possible bias.

UC3.3.2.5: Analyze the impact of the order of the different components
In VisualCit the order of components is fixed and based on the performance and reduction rates of the available components. Further studies are needed to assess the impact of different configurations.

UC3.3.2.6: Enhance human-in-the-loop tools
The current version of VisualCit has an involvement of citizens mainly in setting up the search keywords and in the crowdsourcing. Further involvement with a human.in-the-loop approach can be envisioned in all phases, e.g. based on a partial assessment of a sample to set up thresholds in different phases, based on the GUI to be developed.
This requirement evolves the initial requirement *Requirement 3. Adequate amount of information for citizen scientists*

UC3.3.2.7
Based on the experience of both case studies the need for further tools for being able to analyze large quantities of posts emerges. In addition to building classifiers based on crowdsourcing as from *Requirement 7. Boost crowdsourcing capabilities,* which need to be generalized from the initial work (see details in Section 8), further tools need to include the selection of the tools to be crowdsourced vs the ones to be automatically analyzed.

3.3.3. Decidim4CS new requirements

Thanks to the inherited democratic nature from its decidim base, all comments and actions in Decidim4CS are transparent, and thus visible to the public. On the other hand, like all public platforms on the Internet, as Decidim4CS becomes more prominent, it may suffer from SPAM users and bots. And their actions on the website could create inappropriate or malicious content. Thus, we have defined the following new requirement:

UC3.3.3.1. A citsci needs to provide moderation in debates.
Second requirement we have defined is with respect to the “Deliberation technologies” task that we detail in the next section.

UC3.3.3.2. A citsci needs to export debate data to compute collective decisions

4. Deliberation technologies for citizen science

Every community that tries to find a solution to a problem, scientific or not, needs a way to reach decisions. Deliberation is a commonly resorted process in this regard to identify and weigh the options for the sought decision. Crowd4SDG tackles this need under Task 2.1 - *Deliberation technologies for citizen science*. During this first year, we have addressed two of the subtasks proposed in the Description of Action (DoA), namely: the development of a formal model for large-scale human debates, and the development of algorithms to compute collective decisions. These subtasks are useful to scale-up democratic management to large CS projects. Furthermore, as a result of the work in this task we have also designed, developed, put in production and maintained the Decidim4CS tool. Both lines of work are aligned with SO1.1a: develop software support for structured deliberation.
Since the work on the Decidim4CS has been described in section 2.6, in what follows we detail our advances along the first two more research oriented subtasks.

4.1. A formal model for large-scale human debates

In [Ganzer2020a] we have introduced a new formal model, which we call the relational model, to structure the information in a debate while allowing more expressiveness than existing approaches permit, both in terms of the structure of the debate, and in terms of the opinions expressed by participants. We also propose several methods for aggregating the information of the debate captured by the model to obtain an output reflecting the collective view of the participants involved. Furthermore, we study the formal properties of our aggregation methods with respect to several social choice properties adapted from those proposed in the social choice literature. We follow this path because social choice theory [Aziz2017] is a research field that focuses on establishing the collective opinion of a group facing a choice between many alternatives. Given a set of alternatives and a set of agents who possess preference relations over the alternatives, social choice theory focuses on how to yield a collective choice that appropriately reflects the agents' individual preferences.

Figure 4.1.1. Debate using the relational model (left) and basic elements of the relational model.

Figure 4.1.1 graphically depicts the steps in a debate using our relational model. It also shows the basic elements of our relational model. Next, we summarise the main features of our novel formal model for human debates (thoroughly detailed in [Ganzer2020a]) with respect to the state of the art:

- **Increased expressiveness.** State-of-the-art collective decision-making frameworks based on traditional argumentation models usually take as a starting point some fixed argumentation structure that only models attack relationships between arguments or a combination of attack and support/defence relationships between arguments. These frameworks then allow participants to express opinions about the different arguments included in the debate. The fixed nature of the argumentation structure, even if it is defined by the participants, represents a significant drawback for participation systems. To overcome this problem, our relational model uses relationships that are not subjectively classified as attacks or defences, but only represent the connections between elements of the debate. The subjective classification is applied individually by each participant not in terms of attack or defence, but in terms of acceptability of the connections. Thus, the structure of the

debate is focused on organising relevant information, not on expressing the subjective opinion of the participants, which is added separately using other tools.

- **Going beyond abstract argumentation.** Several approaches in the literature make use of abstract argumentation frameworks, or some variations to represent the elements of a debate. In such frameworks, whole arguments are the atomic elements. In argumentation research, this limitation has led to work on “structured” or “rule-based” argumentation, which constructs arguments out of lower level components like facts and rules. Since we believe that a debate can hinge on being able to address such lower level components, we take a similar, but more general, approach. We construct debates from two types of abstract objects: statements, which represent sentences without reasoning, and the relationships between statements, which represent the existing reasoning connecting the statements. We make a sharp distinction between these two types of information, statements and relationships connecting statements, to allow them to be subsequently evaluated in different manners by the participants in a debate.
- **Compound and real valued opinions.** Previous work on argumentation-based approaches has only allowed participants in a debate to express opinions about either the arguments, or about the relationships between arguments. Our relational model for debates allows participants to provide opinions on both statements and the relationships between them. Opinions about relationships capture participants' acceptance, or otherwise, of the reasoning that the relationship represents; and opinions about statements reflect participants' satisfaction with the statement itself. Furthermore, we allow opinions about relationships and statements to be expressed using real values rather than -discrete values. This feature allows the participants to express their opinions in a wider range of values, making the approach more flexible. (It should be noted though that most existing participation systems just allow users to express agreement or disagreement.)
- **A more flexible notion of coherence.** Previous work on determining collective opinions makes use of a notion of “rationality”' in which an opinion is either determined to be acceptable or not acceptable (where “acceptable” has different interpretations but reflects the constraints on distributions of opinions across statements). We think that this is somewhat limiting. Since the opinions originate with human participants, and humans are not always consistent in their views, we feel that insisting on this rigid form can lead to losing valuable information. Hence, we propose a less restrictive notion of rationality, which we call “coherence”, to assess the degree to which an opinion is coherent, be it from an individual or from the collective aggregation.
- **Aggregation functions exploiting dependencies.** We propose several opinion aggregation functions that use the participants’ opinions on a debate to compute a collective opinion. These proposed functions assume different forms of using the dependencies between opinions to combine them. We provide two families of functions that include a function that does not use dependencies at all, and two other functions that use dependencies rather differently. These families of functions collectively span the ways in which the dependencies can be taken account of, thus making it possible to choose a specific degree of use of the dependencies.
- **Formal analysis.** We study our family of aggregation functions against a wide-range set of social choice properties designed to provide a detailed characterization of their behaviour. We use several properties adapted from the social choice literature to fit our model in order to characterise the formal features of our aggregation functions. We carry out the same study in four scenarios that consider different assumptions on the feature of the participants' opinions.

4.2. Algorithms to compute collective decisions

In [Ganzer2020b] we provide a publicly-available implementation of the algorithm for computing collective decisions specified in [Ganzer2020a], together with all the aggregation operators to aggregate opinions over arguments that are defined in the paper. Furthermore, we have empirically analysed the computational time required by our implementation to compute collective decisions, as thoroughly detailed in [Ganzer2020a]. Our purpose has been to determine whether our approach can indeed handle collective decision-making in practice. In what follows we summarise our study in [Ganzer2020b].

First, we have synthetically generated instances of large-scale human debates, which required the synthetic generation of arguments and opinions on arguments. On the one hand, we artificially generated debates whose arguments are the nodes of a directed acyclic B-hypergraph and whose hyperedges represent the relationships between sentences. We chose the number of sentences to represent small, medium, and large scenarios (within {100,150,200} respectively). Besides that, we also considered that arguments in debates can be more or less connected. Thus, we generated relationships between arguments to represent from low-density debates (single connections from argument to argument) to high-density debates (large number of connections between arguments).
To complete the synthetic generation of debates, we artificially generated opinions for each debate. Thus, the number of opinions for each debate is picked from $\{10^6, 3{\cdot}10^6, 5{\cdot}10^6\}$ to represent the size of the largest known actual-world debates. These numbers of opinions are indeed larger than the most likely scenarios in CS, meaning that the computational times for the application in CS will be much more manageable. To the best of our knowledge, the Brexit discussion on UK [Petitions2019] constitutes the largest such discussion: news outlets reported when the number of supporters passed 2 million [BBC2019], and the numbers kept growing during the 6 month period that the discussion was open. By the time it closed, there were 6,103,056 participants [Petitions2019]. Contrasting numbers of participants can be found for other popular initiatives such as an environmental proposal in Parlement et Citoyens, which had 51.493 votes [Parlement2015], and in the participatory budgeting process in Helsinki [Helsinki2021] with 54.246 registered people, which represents 10% of the city voters. Note that the Parlement et Citoyens and Helsinki debates are probably more representative of real online debates than the Brexit example, where participants were, in effect, just voting on a specific proposal.

We analysed the computational time required to compute collective decisions through two types of analysis:

- sensitivity to the number of participants
- sensitivity to the density of relationships between arguments

As an example (refer to full details in [Ganzer2020a]), Figure 4.2 below shows that the time to compute collective decisions increases as the number of participants increases and the number of arguments (sentences) in the debate increases.

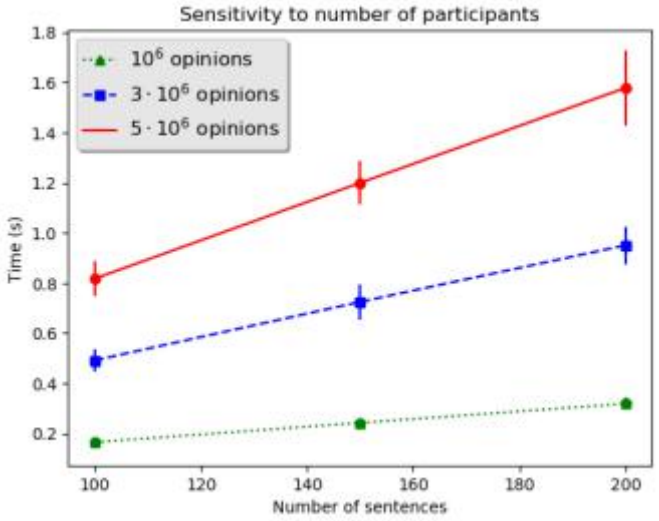

Figure 4.2. Computational time as the number of participants and arguments (sentences) grow.

Overall, notice that our empirical analysis indicates that computing collective decisions in all the artificially generated debates took less than 1.6 seconds. Therefore, we can conclude that the algorithms specified in [Ganzer2020a] and made publicly available in [Ganzer2020b] can be employed to cope with large-scale human debates in real-time.

5. Human-machine collaborative learning

Human-machine collaboration is a model where humans and intelligent systems work together to enhance each other's strengths to solve a given problem. The objective of Task 2.2 - *Human-machine collaborative learning* is the development of open-source software tools and supporting methodologies which enable CS projects to take advantage of the state-of-the-art human-machine collaborative learning algorithms.

The work developed during the first 12 months has followed two different lines. Since Task 2.2 requires integrating with the Pybossa open source software, a first line of work has concentrated on analyzing the current status and prospective evolution of the Pybossa project, to select the most adequate way in which to integrate human-machine collaborative learning. A second line of work has concentrated on building a conceptual and mathematical foundation for human-machine collaborative learning in Citizen Science.

5.1. Human-machine collaborative learning support for Pybossa

PyBossa is a free, open-source crowdsourcing and micro-tasking platform. It enables people to create and run projects that utilise online assistance in performing tasks that require human cognition such as image classification, transcription, geocoding and more. PyBossa is there to help researchers, civic hackers and developers to create projects where anyone around the world with some time, interest and an internet connection can contribute. Pybossa is the core of the Project Builder tool from the CSSK. Therefore, improvements to Pybossa regarding human-machine collaboration will ultimately enhance Project Builder for the same goal.
As a first step in the improvement of the Pybossa suite, we made a deep analysis of the current status of the Pybossa project. This has included a thorough revision of the architecture of Pybossa itself, identification of its key components and their interfaces, and opportunities for extension. Then, we have identified two major functionalities which can move Pybossa towards more intelligent provision of services. The first of them is the automatic computation of intelligent consensus. The second one is intelligent scheduling.

5.1.1. Pybossa strategic direction

This process has also included meeting and online discussion with Daniel Lombraña, the lead developer of Pybossa, to analyze how our projected modifications in Pybossa can benefit the whole Pybossa community. From those meetings we have arrived to the conclusion that currently the strategic direction of the Pybossa project aims towards a reduction in user interface functionality and specialization of the platform as a back office tool for industrial crowdsourcing deployments. This strategic direction severely influences the way in which we can rely and collaborate with the Pybossa infrastructure. Whilst initially the Crowd4SDG project was thought to be contributing code to the core Pybossa project, right now we consider it more likely to incorporate Pybossa just as a tool and build our developments as independent software components.

5.1.2. Computation of intelligent consensus in Pybossa

A recurring theme for citizen scientists using crowdsourcing is how to reach a consensus answer that takes into account the responses of different workers / volunteers, and there is a large literature for this task. However, as of now, Pybossa misses a component that computes consensus responses from its tasks. We have identified this as an opportunity to incorporate intelligent models inside Pybossa. This, together with requirements from Task

2.3 has led to the development of the Crowdnalysis library (explained in section 2.4.). In its current status, Crowdnalysis allows a citizen scientist to download data from a Pybossa CS project and automatically obtain the consensus of the answers reported by the workers/volunteers. We expect to be able to directly integrate this functionality into our custom version of Pybossa in the next release of the CSSK in month 24.

5.1.3. Intelligent scheduling in Pybossa

When a citizen scientist creates a project in Pybossa, he/she selects how different tasks are going to be sent to the workers. This is known as the scheduler. Currently, the available schedulers in Pybossa only take care of sending each task to a fixed number of different workers, disregarding the characteristics of workers and tasks. We have identified this as another opportunity to integrate AI functionality in Pybossa. During this period we have drafted the design of how that functionality could be provided (see Figure 5.1.3). This requires the implementation of a software component (the so called Crowdinator) that (i) continuously monitors the progress of the annotation process in Pybossa, (ii) incrementally computes consensus and characterizes tasks and workers (relying on Crowdnalysis), and (iii) uses that information effectively to perform a better scheduling. During this period we have started working in the design of Crowdinator. We expect to be able to have a first operational version in the next release of the CSSK in month 24.

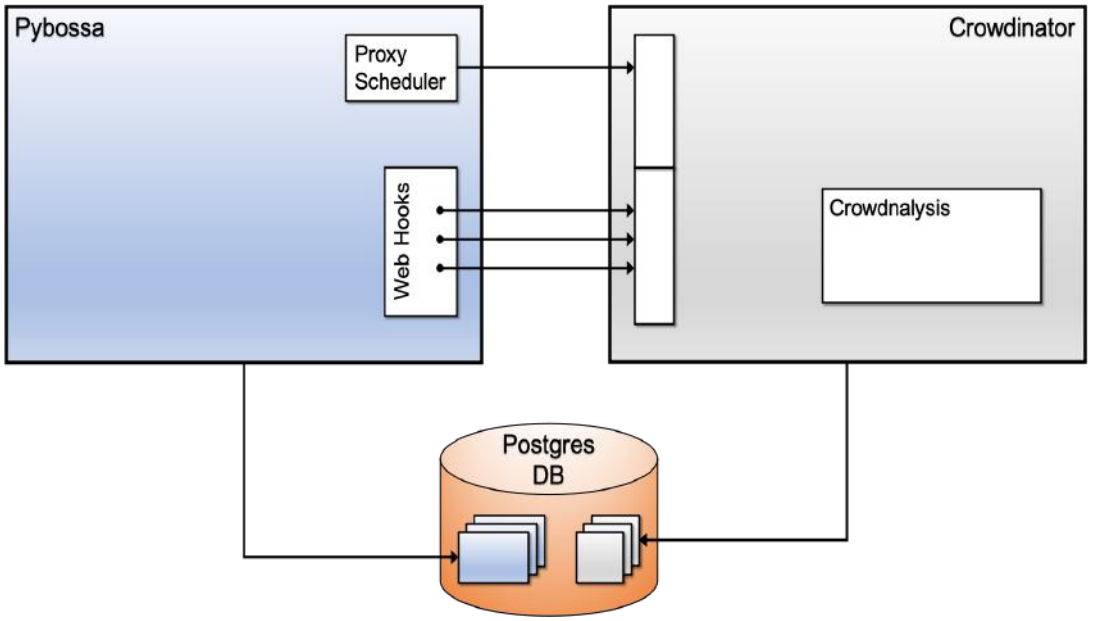

Figure 5.1.3. Providing Pybossa with intelligent scheduling.

5.2. Conceptual and mathematical modeling

As a first step towards building a human-machine collaborative learning platform, we have started performing the data modeling which underlies our analysis and developments. The abstract mathematical data model relies on three main concepts. For clarity, below we exemplify our model in the context of disaster damage assessment by citizen scientists who label images obtained from Twitter. The main concepts of our conceptual model are:

- **Worker:** A worker is any of the participants in the annotation process. In our example, each of the volunteer citizen scientists involved in labeling images is a worker.
- **Task:** A task can be understood as the minimal piece of work that can be assigned to a worker. Labeling each of the images obtained from Twitter is a task.
- **Annotation:** An annotation is the result of the processing of the task by the worker. An example of annotation in the above described disaster management example conveys the following information: *Task 22 has been labeled by worker 12 as "moderate damage"*.

From this data model we build a general probabilistic model which is the main tool that will be used for making sense of the crowd-sourced data. This model is depicted in Figure 2.4.1.2. The fundamental idea is that this probabilistic model will enable us to answer a set of different questions of interest for human-machine collaborative learning. For example:

1. Provided a task: which of the available workers will annotate it better?
2. Provided a worker: which of the available tasks should he/she annotate in order to contribute most to the project?
3. Provided a set of annotations of a set of tasks by a set of workers:
    - How do we create a consensus on the real labels of the images?
    - Which workers are more competent?
    - Which tasks are easier and which are harder?

We have started by tackling the third of these points during the first year of the project and have done so by implementing the Crowdnalysis library which is described in Section 2.4.

A more detailed description of the data model and probabilistic model can be seen in the journal paper that has been published in the high quality (JCR first quartile) Mathematics journal [Cerquides2021].

6. Agreement and data quality analysis

Ensuring data quality in CS projects is fundamental for the reliability of the results inferred from crowdsourced data. The objective of Task 2.3 - *Agreement and data quality analysis* is to evaluate and improve the quality of data provided by CS projects. To fulfill this objective, and according to the task description, we have concentrated our work this period along two lines, both of them clearly specified in the tasks description. In the first line, we have focused on automatically estimating the accuracy of citizen scientists, their responses, and how to best aggregate them. On the second line, we have provided an evaluation of the quality of data collected from social media sources, evaluating its accuracy and precision.

6.1. Automatic estimation of the accuracy of Citizen Science results

This line of work is tightly connected with the computation of intelligent consensus in Task 2.2, since by using the same data analysis models [Jin2020] we can go from the usual output of a crowdsourcing Citizen Science project, namely a set of annotations of a set of tasks by a set of workers and obtain:

- A consensus answer for each of the tasks together with an estimate of the uncertainty of that consensus, as well as an estimate of the difficulty of the task.
- An estimate of the quality, accuracy and potentially a characterization of the typical types of errors for each of the different workers.

As a consequence, in these first 12 months we have focused on creating the technological foundation to be able to effectively perform estimation of accuracy in Citizen Science results, by designing and implementing the Crowdnalysis library. Then we have provided a first evaluation of the Crowdnalysis library in action by using it to analyze the results of a Citizen Science project whose objective was the estimation of severity of the damage observed in images coming from the earthquake that took place in Albania in 2019. More detailed information about the Crowdnalysis library requirements analysis, design and implementation can be found in Section 2.4. As for the Albanian earthquake use case, details can be found in Section 3.2.

6.2. Evaluation of the quality of data collected from social media sources

6.2.1. Assessment criteria

To support users in the selection of the components of a pipeline, in Year 1 we studied a methodology for evaluating a pipeline with a set of criteria to assess a pipeline, described in [Cappiello 2021].

We consider three main criteria for assessing the performance of a pipeline and comparing different configurations: (i) cost, (ii) time, and (iii) quality. We associate these criteria to the single components and study how they can be composed in the global assessment of the pipeline [Cappiello2021].

Cost:
The cost of the execution of the pipeline depends on the amount of resources required for processing all the items of the data set. At the component level, the cost can be computed depending on the type of component we are considering:

- the *computational cost*: in case of automatic components, the cost is related to the computational resources needed for its execution.
- the *crowd cost*: it is the cost of human-in-the-loop components. It depends on the number of items submitted to the crowd. Pricing of crowd tasks is a complex and ethical related issue (see: https://medium.com/ai2-blog/crowdsourcing-pricing-ethics-and-best-practices-8487fd5c9872) that might affect the quality of the result.

Given the cost of each component, the overall cost of the pipeline can be computed as the sum of the costs of each component according to the number of items that each component is going to analyze.

Time:
The time of execution of the pipeline assesses the efficiency of the process. The time depends on the features of the components as well as on the number of items that it has to analyze. Each component is characterised by an average execution time per item. The overall execution time for a sequential pipeline can be computed as the sum of times for each component given the expected number of items to process.

Quality:
Quality is a criteria that assesses the effectiveness of the pipeline. The overall goal is to extract from the original data set, the items that are relevant to the task. More precisely, the quality of the pipeline can be expressed with the following metrics: precision, recall, population completeness, and volume.

Precision can be computed at the component and at the pipeline level. At the component level, a high precision indicates that the component was able to discard not relevant data in its input from its output, thus avoiding waste of resources for processing irrelevant items in the rest of the pipeline. At the pipeline level, high precision indicates that computational and human resources are not going to be wasted in the data analysis part of the pipeline.

Recall assesses the ability to keep items that are relevant to the task.

Population Completeness assesses the amount of data items in the data set of the different classes that the user aims at representing. Population Completeness, measured at the pipeline level, ensures the presence of data items for all the classes of interest. Imbalance in the detection and filtering of different classes is a serious problem since it can lead to *bias* in estimated indicators that are based on the frequency of observed items across the classes. For example, if estimating mask usage in the population from observed images on Twitter, one might underestimate the prevalence of mask usage based on image counts if the mask-detector tends to produce more false negatives (where it miss-classifies images containing individuals wearing masks as not wearing them) than false positives (where it miss-classifies those not wearing masks as wearing them). Estimating the precision and recall of the various components in the pipeline and quantifying the resulting imbalance and its effect on bias of the resulting estimators will be investigated during the second and third year of the project.

Volume is a metric measuring the size of the data set in input or output. Volume, measured at the pipeline level, ensures that sufficient data are available for executing the analysis tasks. The computation of indicators as an output of the data analysis part of the pipeline depends critically on the volume of data available in order to guarantee that uncertainty in the

estimated indicators lies within acceptable levels. Statistical significance measures for quantifying the uncertainty in the estimates will be investigated in the second and third year of the project.

6.2.2. Derivation of statistical indicators and their validation

A second approach for validation of the quality of the results is focusing on the evaluation of indicators obtained from social media pipelines.
For this evaluation an independent source for measuring the same indicators is retrieved, and a statistical analysis of the results is performed.

The following evaluations have been proposed by the project in Year 1:

- analysis of global behavior by country: correlation of results from pipelines with results from survey
- quantification of coverage in terms of number of countries

More work is needed in the direction of evaluating the confidence and uncertainty of the results in order to provide a reliable tool as complementary statistics. e.g. for NSOs.
The first results are illustrated for the COVID-19 pipeline in Section 3.

7. Self-composition. Adaptive services

CSSK tools need to be adaptive to give timely and accurate responses in real-world scenarios. Tackling this need implies a succinct definition and thorough analysis of the components of the tools and their parameters. These goals underlie the main focus of Task 2.4 - *Self-composition. Adaptive services*. As described in the project proposal, in this task "We will start by classifying and characterising existing extraction and filtering tools in order to develop strategies to assess the behaviour of each tool on specific CS projects. We will develop filters to adapt the quantity of information to be analysed to each citizen scientist group's size and data quality requirements. Filters include: reduction of volume, retrieval of related information, mapping information onto domain-specific ontologies. From this, mechanisms for dynamically composing and parametrizing the tools in a CS project will be developed. "

The background material for the project is derived from the H2020 E2mC project, in which Polimi and UNIGE, together with other partners of the consortium developed a basic pipeline to extract visual evidence in emergencies, such as earthquakes and floods, in order to accelerate Copernicus Emergency Management Services [Havas2017]. In Crowd4SDG we generalized the definition of the pipeline and components for a generic VisualCit tool which is not domain specific.

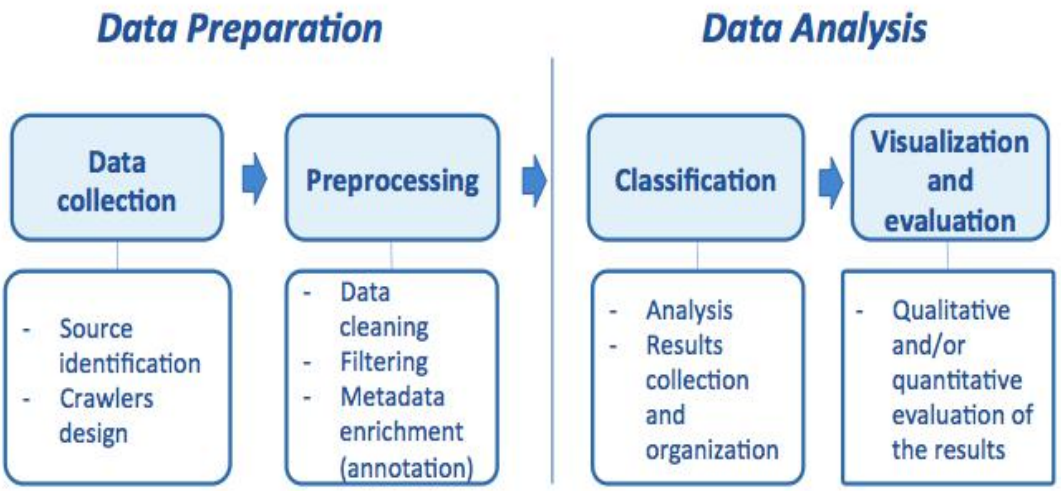

Fig. 7.1 Crowd4SDG VisualCit Basic pipeline and components

The pipeline includes both automatic components and manual activities, to realize a data analysis approach based on the human-in-the-loop approach.

In the first year a common format for the pipeline csv files and naming conventions for the exchanged files have been defined (as reported in the first dataset published in (Pernici 2021)).
A first instantiation of the pipeline has been defined in two case studies illustrated in Section 3.

7.1. Data collection

Social media analysis has been envisioned in the project. Starting from existing components realized in E2mC, a Twitter crawler has been realized, that can be used in two modalities:

- Interactive interface, as a self service, with Twitter credentials from the user of the project;
- Python code that can be run on the machines owned by the pipeline organizers, to distribute computing on different machines.

Future work will focus also on other sources of information for GEAR cycles. A first candidate is Reddit. Information collected with crowdsourcing and available datasets will also be considered.

7.2. Preprocessing

The focus in the first year has been on the classification and realization of different types of generic filters, to select the relevant images from the highly noisy amount of images, in particular from social media like Twitter which contain many duplicates, memes, emoticons, non photos.

The preprocessing components include:

- Cleaning tasks: they aim to (i) identify and correct errors and anomalies (e.g., anomaly detection, imputing missing values); (ii) eliminating duplicates; (iii) eliminating inappropriate and useless data.
- Semantic filters: they reduce the number of data values in order to satisfy the application requirements. Semantic filters select data to consider on the basis of their value or characteristics. For example, if the input source is Twitter, tweets can be selected on the basis of the content (e.g., presence of images) or on the basis of the tweet metadata (e.g., posting time).
- Selectors (or Sampling filters): they are components that reduce the number of values to analyze. In particular, the component is characterized by a reduction rate that is the ratio between the output and the input data sets. In our approach, we use selectors that apply the simple random sampling: elements to be included in the output data set are selected randomly from the input data set. Note that sampling could be also performed using different approaches such as systematic sampling or clustering sampling. Selectors include deduplication components for images, based also on image similarity (using perceptual hash functions).
- Metadata enrichment/Annotation: they extract/add important metadata such as location, topic, image and/or characteristics. Missing metadata are derived from the available information (e.g., derive the location of a tweet from the text) or new metadata are created, such as a classification of the available images into photos and non-photos, like diagrams or memes. Enrichment can be performed automatically or manually, with a human-in-the-loop approach.

7.3. Classification

Classification can be performed manually using crowdsourcing (see the Project Builder in Section 2.3) or automatically, as discussed in Section 8.2.
Classification is the basis for the construction of indicators, including SDG-related indicators (see the COVID-19 case study in Section 3.1).

Crowdsourcing can be used for several purposes at this phase:

- crowdsourcing - for collecting data;
- crowdsourcing - for evaluating the quality of the filters (validation);
- crowdsourcing - for training new filters and improving existing filters using active learning.

The results from crowdsourcing have to be aggregated and evaluated (see Crowdnalysis).

7.4. Visualization and evaluation

This phase is often project specific. In the first year of the project we focused on the following types of visualizations (see VisualCit for COVID-19 in Section 3):

- Visualization of aggregated results;
- Derivation of indicators;
- thematic maps creation.

7.5. Dynamic pipelines

The selection and order of available components in a pipeline for a given case study depends on the goals of the specific case study.
Assessment criteria are based on the criteria defined in Task 2.3.
In [Cappiello2021] we started defining a methodological approach based on goals and constraints defined for a case study and on a characterization of single components. The methodology will be further developed in Year 2 based on Gear 2 experimentation.

8. Enriching Social Media content by Citizen scientist

Social media content constitutes the majority of content on the Web and provides a useful data source for measuring SDG indicators, amplifying the collection capabilities of individual citizen scientists to include the observations and experiences of millions of users of social media platforms. Content on such platforms is notoriously noisy, however, making the manual vetting, cleaning and annotating of the content by citizen scientists a necessity. To this end, Task 2.5 - *Enriching Social Media content by Citizen scientist* aims to benefit from content beyond keyword-based filtering techniques. Specifically, the task focuses on images extracted from social media to provide visual evidence of ongoing events and situations that can enable the derivation of SDG indicators, particularly in areas or topics in which NSOs need complementary data for their analysis. AI techniques are needed to support Citizen scientists activities in order to provide them with an adequate amount of information to be analyzed, avoiding wasting human resources on clerical tasks.

In the first year of the project, we focused on three aspects, derived from the initial requirements, as they are essential as a basis for the results:

- Development of new Machine Learning models that can be provided as filtering components for the social media analysis pipeline to reduce the noise of the social media data. The filtering of the images which are not related to the task is essential in the data preparation phase, for all case studies applying this methodology, as many images are memes or graphics, or not safe for work. Some specific filters have been developed as described in Section 8.1.;
- The automatic construction of new classifiers from crowd annotations. As mentioned in the case study, one of the limiting factors can be the size of the crowd, which is a limited resource. Combining crowdsourcing and the dynamic construction of new classifiers can mitigate this problem;
- The derivation of statistical indicators from the pipeline and their validation. As one of the goals of the project is to derive indicators, we were focusing on producing indicators starting from crowdsourced information.

8.1. AI enhanced filtering components

The concept of irrelevance may vary in each case being considered in the analysis.
In the first year we focused on general-purpose filters that can be applied in a variety of cases. The actual choice of filters to be applied will depend on the pipeline being composed, the problem domain, and the number of posts being analyzed.

Thus we perform a set of filtering operations to extract only those images that are likely relevant. For this purpose, we built an image filtering pipeline based on deep learning techniques, including both state-of-the-art models pre-trained on large public datasets, and custom filters built according to our needs. The pipeline performs the following filter operations:

- Removing non-photos;
- Removing Not Safe For Work (NSFW) content;
- Detecting the scene;
- Detecting people.

Removing non-photos

In order to efficiently remove from the images retrieved from social media with the crawler those that do not represent photos, a photo detector was implemented. Crawled images contained a significant percentage of irrelevant images corresponding to internet memes or modified photos with text. To tackle this problem, a VGG19 model, pre-trained on the ImageNet dataset[3], was fine-tuned on a data set containing 3,376 images labelled as memes / non acceptable photos (taken from the Reddit Memes Dataset (https://www.kaggle.com/sayangoswami/reddit-memes-dataset), and 2,448 images considered acceptable (taken from the Multi-Salient-Object (MSO) Dataset (https://www.kaggle.com/jessicali9530/mso-dataset). To fine-tune the algorithm, VGG19's last layer was substituted to adapt the model for the new classification task, and all layers inherited from the original architecture are held frozen during training. In this way, the model achieved excellent performance on the filtering task.

Removing NSFW content
It is critical to discard all Not Safe For Work (NSFW) content from our data before feeding it to the crowd workers. To ensure this, we made use of Yahoo's implementation of a NSFW classifier, OpenNSFW.
Deciding what type of content is safe or not is subjective and context-specific. Yahoo's model specifically filters out pornographic content, while it does not address non-photos or offensive text, which we will both target by using a photo filter. It also does not address images depicting violence, which however we might want to include to investigate people's behaviour to help policy makers. We use this model as a preliminary filter, knowing that it provides a limited guarantee on the accuracy of the output. We will thus necessarily warn our crowd on the probability of facing explicit content and ask to alert us in such a case.

Detecting the scene (Public/Private)
Selecting the right scene in images allows extracting a more meaningful subset of data for our task. For this purpose we introduced a scene detector in our pipeline.
This component consists in a convolutional neural network able to classify an image as belonging to one of a set of scene categories. In our framework we introduced an open-source model pre-trained on Places365, a public dataset of images corresponding to 365 scene categories.
For our specific task we thought a more meaningful distinction was between public and private scenes. Thus, we aggregated the original 365 scenes in these two subsets.

Detecting People
We can greatly benefit from detecting required objects in a scene to narrow down the most relevant images for our purpose. Different domains may have different requirements in this respect. In previous research, focusing on selecting images for giving evidence in emergency situations, it was noted that filtering out images with faces would increase the relevance percentage of the posts (Barozzi et al., 2019). In the case study performed in the first year of the project, instead, the attention was focusing on behaviours of persons in a social setting, so detecting persons is important to select relevant posts. In general, in future developments it is useful to be able to select objects relevant for the case study at hand.
For this purpose we introduced in our pipeline the YOLO (You Only Look Once) object detector, pre-trained on the COCO (Common objects in context) dataset.
In the specific scenario of gathering relevant images for policy makers during the COVID-19 outbreak developed in the first year of the project, we extracted images containing people. In

[3] References to specific tools and models are provided in the published paper [Negri2021].

addition, filtering images with at least two people showed a significant increase of the quality of the result, for example discarding selfies.

8.2. Automatic construction of new classifiers from crowd annotations

As one of the critical aspects in VisualCit pipelines is the ability of processing preprocessed data with crowdsourcing, due to the large volume of data in some case studies.
In Year 1 of the project we started investigating the automatic training of classifiers starting from crowdsourced data, as shown in Fig. 6. The goal is to be able to evaluate selected images both manually and automatically with a classifier that is constructed during the pipeline execution.

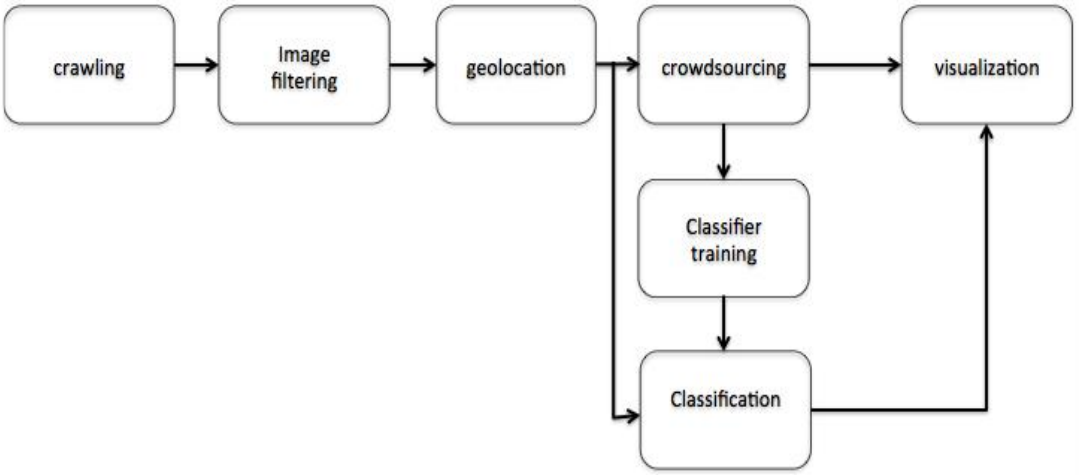

Fig. 8.1 - AI-enhanced pipeline

The first results are reported in [Scuratti 2021a, Scuratti 2021b].

9. Interaction of WP2 with other Crowd4SDG work packages

WP2 plays a central role in the Crowd4SDG project as depicted in Figure 1.3. The CSSK created by WP2 partners are primarily used in the challenge-based innovation projects initiated by WP3. The data regarding the social and economical impact of these bottom-up CS projects are then analysed by WP4 and WP5. Specifically, WP4 measures the analytics of citizen collaborations (e.g. team diversity and organisational structure) leveraging digital traces of the usage of the CSSK tools. WP5, on the other hand, conducts a comparative analysis of the statistical data to meet the quality requirements applied by NSOs. The analysis of WP4 and WP5 provides feedback to WP2 for a further refinement of the tools, and also helps all project partners to improve the GEAR methodology itself.

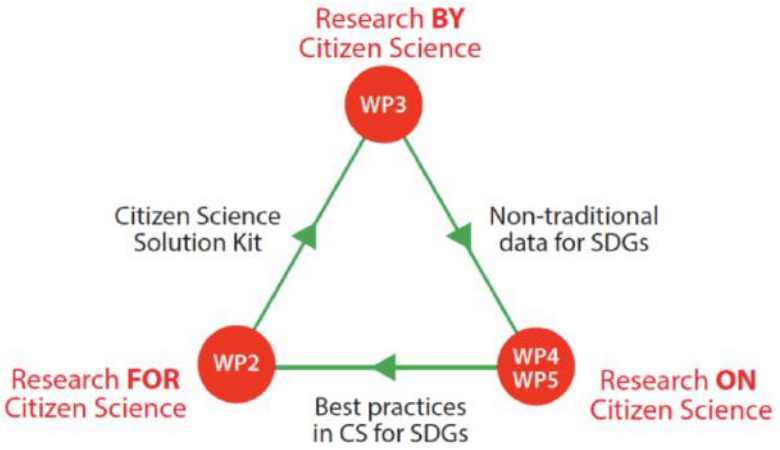

Figure 1.3. Crowd4SDG Research Triangle

9.1. Connection with WP5

WP5 partners analysed the datasets utilized by CSSK tools in the first year and the results were presented to several NSOs and major stakeholders in the panel "Unleashing the potential of Citizen Science Data for monitoring the SDGs" hosted by UNITAR on May 27th. There was a wide attendance of more than 100 people and the tools received positive feedback. One of the three datasets meets the criteria to be used by NSOs as-is, and WP5 identified the needs for the other two datasets which can easily be fulfilled as the needs are basically about missing documentation. We consider this quite a positive step forward for the project, since it proves that CS generated data can fulfill the needs of NSOs and clears the path towards real-life use of the tools and methodologies developed in Crowd4SDG. Detailed analysis report can be found in the deliverable D5.2.

9.2. Connection with WP3

In the first GEAR cycle, in alignment with the objectives of WP2 to support citizen scientists, the tools were presented to all participants of the O17 challenge in a webinar. Special emphasis was put on Project Builder, SDG in Progress and Decidim4CS as these tools might be of first use for the teams in this phase (i.e. Evaluate) of the GEAR cycle.

Beside the presentation, the Tools page of the project website was prepared to describe the tools and provide user stories which highlight examples of their use cases. Furthermore, we are maintaining a dedicated website for the CSSK at crowd4sdg.github.io.

As for the Accelerate phase, the CBI workshop organizers re-informed the selected teams (after O17) about the tools and to whom they should contact if they needed further information.

Among the different tools in the CSSK, SDG in Progress has been used by all teams in the first GEAR cycle. It served the teams to document their projects. Project Builder was tried by one team, whereas the planned usage of Decidim4CS was documented in another team's final pitch. Other tools like VisualCit and Crowdnalysis have not been incorporated by the participants of the first cycle. We have observed a clear correlation between the use of a tool and its mandatory status during the GEAR cycle. According to that, as a consortium we have agreed on making the usage of at least one of the data generating tools mandatory per team for the second GEAR cycle, to foster its usage by participants of the GEAR cycle.

On the other hand, to increase the user base outside the GEAR cycle, we have also started marketing our tools to users in different potentially interested communities. For example, hands-on demonstrations for Project Builder, VisualCit and Decidim4CS were presented online at the Geneva Tsinghua Initiative (GTI) Master Workshop (gt-initiative.org) that was held on March 25-26th. The tools were presented to more than 20 international students of 5 teams by the partners who are developing these tools. The students were then asked to use the tools, receiving hands-on help, and their feedback was noted. The impact of the outreach activity is clearly observable in the example of Decidim4CS. After the workshop, the geographical spread of the usage of the tool increased from 4 to 10 countries from Europe, Asia, and North and South America, as reported by Google Analytics for the website.

A survey was conducted among the GTI Workshop students, and 11 of them participated. The aggregated results of this survey are given in Table 1.3. Being precautious about an early generalization, we see that although the CSSK tools are fairly intuitive to understand, they may not be easily applicable to ongoing projects. The reason for this probably lies in the free text feedback from the students. There was a consensus among them that they could have chosen their topics adequately where these tools could be relevant, if they had an earlier introduction. And they agreed also that this introduction would also give them more time to think of the possibilities of incorporating these tools in their projects, instead of coming up with an idea during a short demo period. Thus, Table 1.3 also justifies the consortium's decision mentioned above on the promotion of the tools to the GEAR 2 cycle participants in a timely fashion.

	Project Builder	VisualCit	Decidim4CS
Do you now have a clear picture on what the presented tool is used for?	2,2	2,1	2,4
Do you now see yourself using this tool for your project?	0,9	0,9	1,4

Table 1.3. Aggregated answers of the GTI Workshop participants on the scale: (3) Yes, Very well; (2) Yes, Somewhat; (1) Not Sure; (0) Not at all.

10. Conclusions and future work

In this deliverable we have provided a summary description of the work developed in the first 12 months of the project. Here, we succinctly report the main conclusions and future work avenues.

- Citizen Science Solution Kit (CSSK): We have created a first version of the CSSK by enhancing existing open source CS tools with artificial intelligence to analyse social media which has been able to effectively process large quantities of posts (thousands). In the following 24 months we expect to be able to scale up VisualCit to very large quantities of posts. Thus, we are progressing towards SO1.1: “Enhance existing open source CS tools with artificial intelligence to analyze social media and other non-conventional big data sources, in particular for monitoring the impacts of extreme climate events, to be able to effectively process very large quantities of posts (millions) when the events are still in progress.”, and towards SO1.1.e: ”Reduce the knowhow and resources needed by citizens wanting to start in grassroots initiatives to create new CS projects”. Since some of the tools in the CSSK (i.e. Crowdnalysis, VisualCit) rely on AI techniques) we are also progressing towards SO1.1f: “Provide a CS solution kit using advanced AI techniques”.
- Task 2.1 - Deliberation technologies for citizen science: Out of the three subtasks that appear in the task description, we have covered two, namely “A formal model for large-scale human debates”, and “Algorithms to compute collective decisions". In the remaining 8 months of this task we plan to work on the third one “A formal model for large-scale human-machine debates”. Furthermore, we have developed and deployed Decidim4CS, a tool to help support structured deliberation in CS projects. This work makes progress towards objective SO1.1.a: “Develop software support for structured deliberation”.
- Task 2.2 - Human-machine collaborative learning: We have designed how to integrate human-machine collaborative learning in the open source tool Pybossa, we have created a conceptual and mathematical model [Cerquides2021], and an open source tool (Crowdnalysis) for the aggregation of crowdsourcing annotations. In [Cerquides2021] we have shown that proper aggregation through Crowdnalysis reduces the amount of effort requested from humans in order to reach an equivalent level of data quality in a CS project. This work is aligned with SO1.1.b: “Set up human-machine collaborative learning strategies to reduce the amount of effort requested from humans". In the following two years we will concentrate on further reducing the amount of effort requested from humans by (i) taking advantage of active learning strategies through the implementation of Crowdinator, and (ii) continuing work on Crowdnalysis.
- Task 2.3 - Agreement and data quality analysis: In Year 1, the focus has been on developing techniques for the assessment of the results of crowdsourcing and in investigating the quality dimensions to be considered for indicators from social media pipelines. The main objective has been *SO1.1c: evaluate and improve the quality of data provided by CS projects*. Ongoing work for the next year is refining the evaluation of missing or Not answered questions by the crowd and the implications of the final results, and in providing techniques to estimate the quality of indicators for SDGs, in general and also focusing on the needs of GEAR 2 projects.
- Task 2.4 - Self-composition. Adaptive services: In Year 1, we have designed and developed a prototype of an image-based social media pipeline (VisualCit), which extracts and prepares social media posts for crowdsourcing and extracts evidence of event and country-based statistics [Negri2021, Scuratti2021a]. We have also

characterized the components of the pipeline to make it adaptive [Cappiello2021]. This work is aligned with *objectives SO1.1d: combine existing computational techniques for social media data extraction for monitoring SDGs, SO1.1e: reduce the knowhow and resources needed by citizens wanting to start in grassroots initiatives to create new CS projects.* In Year 2 we will focus on providing an easier and visual access to the tools by the citizens, to facilitate human-in-the-loop analysis to integrate automatic analysis of posts. We will continue analyzing the characteristics of the tools for improving the adaptivity of the VisualCit pipeline. In addition, we will support the use of VisualCit inside the selected GEAR 2 projects.

- Task 2.5 - Enriching Social Media content by Citizen scientist: In Year 1 we focused on introducing AI techniques to satisfy some of the initial requirements emerging with the first experimentation. We added to the initial pipeline derived from background knowledge of the E2mC project new components for automatic image filtering to reduce the workload for the crowd and we started to complement the manual analysis performed with crowdsourcing by building automatic classifiers leveraging crowdsourced information. In this way we focused on goals *SO1.1b: set up human-machine collaborative learning strategies to reduce the amount of effort requested from humans, SO1.1f: provide a Citizen Science Solution Kit using advanced AI techniques,* and *SO1.1g: explore social media flows between citizens doing CS projects and AI components.* Other activities planned according to work plan will be developed in Years 2 and 3, in particular using feedback from the crowd during or after an emergency event due to climate change to dynamically improve the performance of the geolocation algorithms (boosting crowdsourcing) and using graph embeddings (specifically geographical locations) to leverage available knowledge in entity linking. On demand, depending on requests from projects in the GEAR cycle, based on already available background tools we will work on exploiting the extraction of textual clues from images.

In conclusion, during this first year of the project we have been able to significantly improve some of the already existing tools in the CSSK, and also to create new tools which make use of Artificial Intelligence. Tighter integration of the tools in the CSSK, more advanced functionalities, and an increased user base are some of the general objectives for the remaining two years of the project.

References

[Aroyo2015] Aroyo, L.; Welty, C. Truth Is a Lie: Crowd Truth and the Seven Myths of Human Annotation. AI Magazine 2015, 36, 15–24. Number: 1, doi:10.1609/aimag.v36i1.2564.

[Aziz2017] H. Aziz, F. Brandt, E. Elkind, and P. Skowron. Computational social choice: The first ten years and beyond. Computer Science Today, 10000, 2017.

[Barozzi2019] Sara Barozzi, Jose Luis Fernandez-Marquez, Amudha Ravi Shankar, Barbara Pernici: Filtering images extracted from social media in the response phase of emergency events. ISCRAM 2019

[BBC2019] BBC News. “Cancel Brexit” petition passes 2m signatures on Parliament site. https://www.bbc.com/news/uk-politics-47652071, 03 2019. last visited 02/2021.

[Cappiello2021] C. Cappiello, B. Pernici, M. Vitali, Modeling Adaptive Pipeline for Crowd Enhanced Processes, *submitted to ER 2021 30/3/2021*

[Carpenter2017] Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., & Riddell, A. (2017). Stan: A Probabilistic Programming Language. Journal of Statistical Software, 76(1), 1 - 32. doi:http://dx.doi.org/10.18637/jss.v076.i0

[Cerquides2021] Cerquides, J.; Mülâyim, M.O.; Hernández-González, J.; Ravi Shankar, A.; Fernandez-Marquez, J.L. A Conceptual Probabilistic Framework for Annotation Aggregation of Citizen Science Data. Mathematics 2021, 9, 875. https://doi.org/10.3390/math9080875

[Dawid1979] Dawid AP, Skene AM. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. Appl Stat. 1979; 28(1):20. doi:10.2307/2346806

[Ganzer2020a] Ganzer, J., Criado, N., Lopez-Sanchez, M., Parsons, S., & Rodriguez-Aguilar, J. A. (2020). A model to support collective reasoning: Formalization, analysis, and computational assessment. arXiv preprint arXiv:2007.06850.

[Ganzer2020b] Ganzer, J., Criado, N., Lopez-Sanchez, M., Parsons, S., & Rodriguez-Aguilar, J. A. (2020). Bitbucket repository, https://bitbucket.org/jariiia/workspace/projects/DRF.

[Gomaa2011] Hassan Gomaa. 2011. Software Modeling and Design: UML, Use Cases, Patterns, and Software Architectures (1st. ed.). Cambridge University Press, USA.

[Havas2017] Clemens Havas, Bernd Resch, Chiara Francalanci, Barbara Pernici, Gabriele Scalia, Jose Luis Fernandez-Marquez, Tim Van Achte, Gunter Zeug, Maria Rosa (Rosy) Mondardini, Domenico Grandoni, Birgit Kirsch, Milan Kalas, Valerio Lorini, Stefan Rüping: E2mC: Improving Emergency Management Service Practice through Social Media and Crowdsourcing Analysis in Near Real Time. Sensors 17(12): 2766 (2017)

[Helsinki2021] City of Helsinki. https://www.hel.fi/helsinki/en/administration/participate/channels/participation-model/, last visited 02 2021.

[Imran2014] Imran, M.; Castillo, C.; Lucas, J.; Meier, P.; Vieweg, S. AIDR: Artificial intelligence for disaster response. Proceedings of the 23rd International Conference on World Wide Web, 2014, pp. 159–162.

[Imran2020] Imran, M.; Alam, F.; Qazi, U.; Peterson, S.; Ofli, F. Rapid Damage Assessment Using Social Media Images by Combining Human and Machine Intelligence. arXiv preprint arXiv:2004.06675 2020

[Jin2020] Jin, Yuan, Mark Carman, Ye Zhu, and Yong Xiang. "A Technical Survey on Statistical Modelling and Design Methods for Crowdsourcing Quality Control." Artificial Intelligence 287 (October 1, 2020): 103351.

[Negri2021] V. Negri, D. Scuratti, S. Agresti, D. Rooein, G. Scalia, J L. Fernandez-Marquez, A. Ravi Shankar, M. Carman and B. Pernici, Image-based Social Sensing: Combining AI and the Crowd to Mine Policy-Adherence Indicators from Twitter, accepted at ICSE. Track Software Engineering in Society, May 2021 https://arxiv.org/abs/2010.03021

[Papastamoulis2016] Papastamoulis P. label.switching: An R Package for Dealing with the Label Switching Problem in MCMC Outputs. Journal of Statistical Software, Code Snippets, 2016, 69(1): 1-24.

[Parlement2015] Parlement & Citoyens. Draft law for the recovery of biodiversity, nature, and landscapes. https://parlement-et-citoyens.fr/consultation/projet-de-loi-pour-la-reconquete-de-la-biodiversite-de-la-nature-et-des-paysages/presentation/presentation-et-suivi-6, 12 2015. last visited 02/2021.

[Pernici2020] Barbara Pernici, CROWD4SDG: Crowdsourcing for sustainable developments goals, 248-253, in Book of Short Papers, SIS 2020, Pearson, 2020 link

[Pernici2021] Barbara Pernici. (2021). Crowd4SDG-VisualCit COVID-19 behavioral indicators (Version 1.0) [Data set]. Zenodo. http://doi.org/10.5281/zenodo.4539697

[Petitions2019] Petitions: Brexit. "Revoke Article 50 and remain in the EU" at Petitions UK Government and Parliament. https://petition.parliament.uk/archived/petitions/241584, 08 2019. last visited 02/2021.

[RaviShankar2019] Ravi Shankar, A.; Fernandez-Marquez, J.L.; Pernici, B.; Scalia, G.; Mondardini, M.R.; Di Marzo Serugendo, G. Crowd4Ems: A crowdsourcing platform for gathering and geolocating social media content in disaster response. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 2019, 42, 331–340.

[Scalia2020] G. Scalia, C. Francalanci, B. Pernici, CIME: Context-aware geolocation of emergency-related posts, under revision for Geoinformatica, 2020

[Scuratti2021a] D. Scuratti, M. Carman, B. Pernici, Boosting crowdsourcing capabilities, *submitted for publication, 2021*

[Scuratti 2021b] D. Scuratti, The use of Crowdsourcing for Mining Policy Adherence Indicators from Social Media, Master's Thesis, Politecnico di Milano, April 2021

[Stephens2000] Stephens, M.. Dealing with label Switching in mixture models. Journal of the Royal Statistical Society Series B, 2000, 62, 795-809.

[Zahra2020] Zahra, K., Imran, M., Ostermann, F.O.: Automatic identification of eyewitness messages on Twitter during disasters. Inf. Process. Manag. 57(1) (2020)

Annex : List of abbreviations

Abbreviation	Description
AI	Artificial Intelligence
CBI	Challenge-based Innovation (in-person coaching)
CoSo	COllaborative SOnar
CS	Citizen Science
CSSK	Citizen Science Solution Kit
DoA	Description of Action
GA	Grant Agreement
GEAR	Gather, Evaluate, Accelerate, Refine
GUI	Graphical User Interface
GTI	Geneva Tsinghua Initiative
ML	Machine Learning
NSO	National Statistical Office
O17	Open Seventeen Challenge (online coaching)
SDG	Sustainable Development Goal
SO	Specific Objective in GA
WP	Work Package