# beAWARE

Enhancing decision support and management services in extreme weather climate events

700475

# D3.4

# Advanced techniques for content distillation from multilingual textual and audiovisual material

| | |
|---|---|
| **Dissemination level:** | Public |
| **Contractual date of delivery:** | Month 34, 30 October 2019 |
| **Actual date of delivery:** | Month 35, 30 November 2019 |
| **Workpackage:** | WP3 Early warning generation |
| **Task:** | T3.1 Crisis Classification<br>T3.2 Concept and conceptual relation extraction from textual information<br>T3.3 Concept and event detection from multimedia |
| **Type:** | Report |
| **Approval Status:** | Draft |
| **Version:** | V1.0 |
| **Number of pages:** | 71 |
| **Filename:** | D3.4_beaware_advanced_techniques_for_content_distillation_from_multilingual_textual_and_audiovisual_material_2019-11-30_v1.0 |

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 700475

**Abstract**

The deliverable describes the final versions of the analysis modules of the beAWARE system and presents technical evaluations of each of the components involved in extracting contents from audio, text and multimedia materials pertinent to the third pilot.

The information in this document reflects only the author's views and the European Community is not liable for any use that may be made of the information contained therein. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

Co-funded by the European Union

This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 700475

# History

| Version | Date | Description / Reason of change | Revised by |
|---|---|---|---|
| V0.1 | 20/9/2019 | Deliverable outline | All consortium |
| V0.2 | 19/10/2019 | IBM contributions | IBM |
| V0.3 | 21/10/2019 | First draft of common parts | CERTH, UPF |
| V0.4 | 27/10/2019 | UPF contributions | UPF |
| V0.5 | 18/11/2019 | Complete document submitted for the 1st internal review | UPF, CERTH, IBM |
| V0.6 | 21/11/2019 | 1st Internal review | IOSB |
| V0.7 | 25/11/2019 | Updated document, submitted for the 2nd internal review | UPF, CERTH, IBM |
| V0.8 | 27/11/2019 | 2nd Internal review | IOSB |
| V0.9 | 29/11/2019 | Final Version | UPF, CERTH, IBM |

# Author List



| Organisation | Name | Contact Information |
|---|---|---|
| UPF | Gerard Casamayor | gerard.casamayor@upf.edu |
| UPF | Simon Mille | simon.mille@upf.edu |
| UPF | Alexander Shvets | alexander.shvets@upf.edu |
| IBM | Benjamin Mandler | MANDLER@il.ibm.com |
| CERTH | Antzoulatos Gerasimos | gantzoulatos@iti.gr |
| CERTH | Krestenitis Marios | mikrestenitis@iti.gr |
| CERTH | Mavropoulos Thanasis | mavrathan@iti.gr |
| CERTH | Michail Emmanouil | michem@iti.gr |
| CERTH | Orfanidis George | g.orfanidis@iti.gr |



# Reviewer list

| Organisation | Name | Contact Information |
|---|---|---|
| IOSB | Hellmund Tobias | tobias.hellmund@iosb.fraunhofer.de |
| IOSB | Blume Jan-Wilhelm | jan-wilhelm.blume@iosb.fraunhofer.de |

# Executive Summary

This deliverable reports on the advanced techniques for concept and conceptual relation extraction from multimedia and textual content. These techniques are implemented in the final version of the modules used for the 3rd pilot of the beAWARE project. Despite the heterogeneous nature of the inputs processed by the analysis modules, the information produced by them uses the same ontological representation developed as part of T4.3 (see D4.2). This facilitates the semantic integration of distilled contents in the project KB.

D3.4 reports advances in tasks T3.1 (Crisis classification), T3.2 (Concept and conceptual relation extraction from textual information) and T3.3 (Concept and event detection from multimedia). The following modules of the beAWARE system covered by this report: Audio Analysis, Image analysis, Video Analysis, Drones Analysis, Crisis Classification and Text Analysis. The Drones Analysis module is a new addition to the system and is the only module that was not present in D3.3.

In terms of inputs to the system, the modules listed analyze audio, images and videos sent by first responders and citizens using the beAWARE mobile app, footage received from drones, and social media posts received by the Social Media module.

CERTH was responsible for the development of methods for T3.1 and T3.3, and the development of Audio Analysis, Image analysis, Video Analysis and Crisis Clasification. CERTH and IBM have contributed towards the design and implementation of the Drones Analysis module. UPF was responsible for the development of methods to T3.2 and the development of the Text Analysis module.

In addition to describing the advanced methods developed for the tasks and the final versions of the modules, this document also reports technical evaluations for each one of the components involved and presents and discusses the results. These evaluations are aimed at assessing the scientific contribution of each component to their respective research fields but using materials relevant the project and more specifically to the 3rd pilot.

# Abbreviations and Acronyms

| | |
|---|---|
| **AA** | Activity Areas |
| **API** | Application Programming Interface |
| **ASR** | Automatic Speech Recognition |
| **BoW** | Bag-of-Words |
| **CAP** | Common Alert Protocol |
| **CCTV** | Closed Circuit TeleVision |
| **CNN** | Convolutional Neural Networks |
| **CRF** | Conditional Random Field |
| **CWRT** | CrossWords Reference Templates |
| **DDP-HMM** | Dependent Dirichlet Process-Hidden Markov Model |
| **DRSs** | Discourse Representation Structures |
| **DUL** | Dolce + DnS Ultralite |
| **EL** | Entity Linking |
| **EM** | Expectation Maximization |
| **EmC** | Emergency Classification |
| **EmL** | Emergency Localization |
| **EM** | Expectation Maximization algorithm |
| **FC** | Fully Connected |
| **FN** | False Negative |
| **FP** | False Positive |
| **Fps** | Frames per second |
| **GMM** | Gaussian Mixture Model |
| **GMMs** | Gaussian Mixture Models |
| **GPD** | Generalized Probabilistic Descent |
| **GPR** | Gaussian Process Regression |
| **HMM** | Hidden Markov Model |
| **HoGP** | Histograms of Grassmannian Points |
| **HOOF** | Histograms of Oriented Optical Flow |
| **IoU** | Intesection over Union |
| **IRIs** | Internationalized Resource Identifies |
| **KB** | Knowledge Base |

| | |
|---|---|
| **KCF** | Kernelized Correlation Filters |
| **LAS** | Labeled Attachment Score |
| **LBPs** | Local Binary Patterns |
| **LDS** | Linear Dynamical Systems |
| **LDT** | Linear Dynamic Texture |
| **LOD** | Linked Open Data |
| **MAP** | Maximum a Posteriori Adaptation |
| **MCE** | Minimum Classification Error |
| **MCMC** | Markov Chain Monte Carlo |
| **MFCC** | Mel-frequency cepstrum coefficients |
| **MLLR** | Maximum Likelihood Linear Regression |
| **MLP** | Multi-Layer Perceptron |
| **MST** | Minimum-Spanning Tree |
| **MTA** | Multilingual Text Analysis |
| **NER** | Named Entities Recognition |
| **NEs** | Named Entities |
| **NLP** | Natural Language Processing |
| **NN** | Neural Network |
| **ObD** | Object Detection |
| **OWL** | Web Ontology Language |
| **PCA** | Principal Component Analysis |
| **POS** | Part-Of-Speech tagging |
| **PSAP** | Public Safety Answering Point |
| **PTB** | Penn Treebank |
| **RDF** | Resource Description Format |
| **ROI** | Region Of Interest |
| **SLIC** | Simple Linear Iterative Clustering |
| **SoA** | State of the Art |
| **STOEF** | Spatio-Temporal Oriented Energy Features |
| **SVM** | Support Vector Machine |
| **TN** | True Negative |
| **TP** | True Positive |
| **UAS** | Unlabeled Attachment Score |

| | |
|---|---|
| **UAV** | Unmanned Aerial Vehicles |
| **UD** | Universal Dependency |
| **URL** | Uniform Resource Locator |
| **VLBP** | Volume Local Binary Patterns |
| **VQ** | Vector Quantization |
| **WSD** | Word Sense Disambiguation |

# Table of Contents

## List of Figures

## List of Tables

# 1 Introduction

This deliverable describes the advances in WP3 tasks within the months 24 to 36 of the beAWARE project. It covers tasks T3.1 (Crisis Classification), T3.2 (Concept and conceptual relation extraction from textual information) and T3.3 (Concept and event detection from multimedia). These tasks contribute to the 4th milestone MS4 “Final Prototype” corresponding to the final SW development cycle of the project, as shown in Figure 1. Deliverable D3.4 follows deliverable D3.3, which presented the results of the 1st milestone MS3 “First prototype”, and deliverable D7.6, which presented the results of the 2nd milestone MS2 “Second prototype”. First and second protoypes correspond to the 1st (heat scenario) and 2nd (flood scenario) pilots of the project, while the final prototype described in this document corresponds to the 3rd pilot (fire scenario).

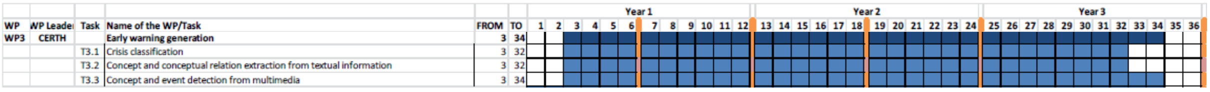



Figure 1: WP3 tasks and timeline

With regards to the system architecture, the modules covered by this report -Audio Analysis, Image analysis, Video Analysis, Drones Analysis, Crisis Classification and Text Analysis- are all shown as part of the Data Analysis & Processing section of Figure 2, which depicts the overall architecture of the beAWARE system.

T3.1, T3.2 and T3.3 of WP3 interact with almost all other WPs, especially with the tasks in WP4 - Aggregation and semantic integration of emergency information for decision support and early warnings generation, but also with tasks in WP5 - Multilingual report generation, WP6 - Main Public Safety Answering Point for emergency multimedia enriched calls and WP7 - System development, integration and evaluation.

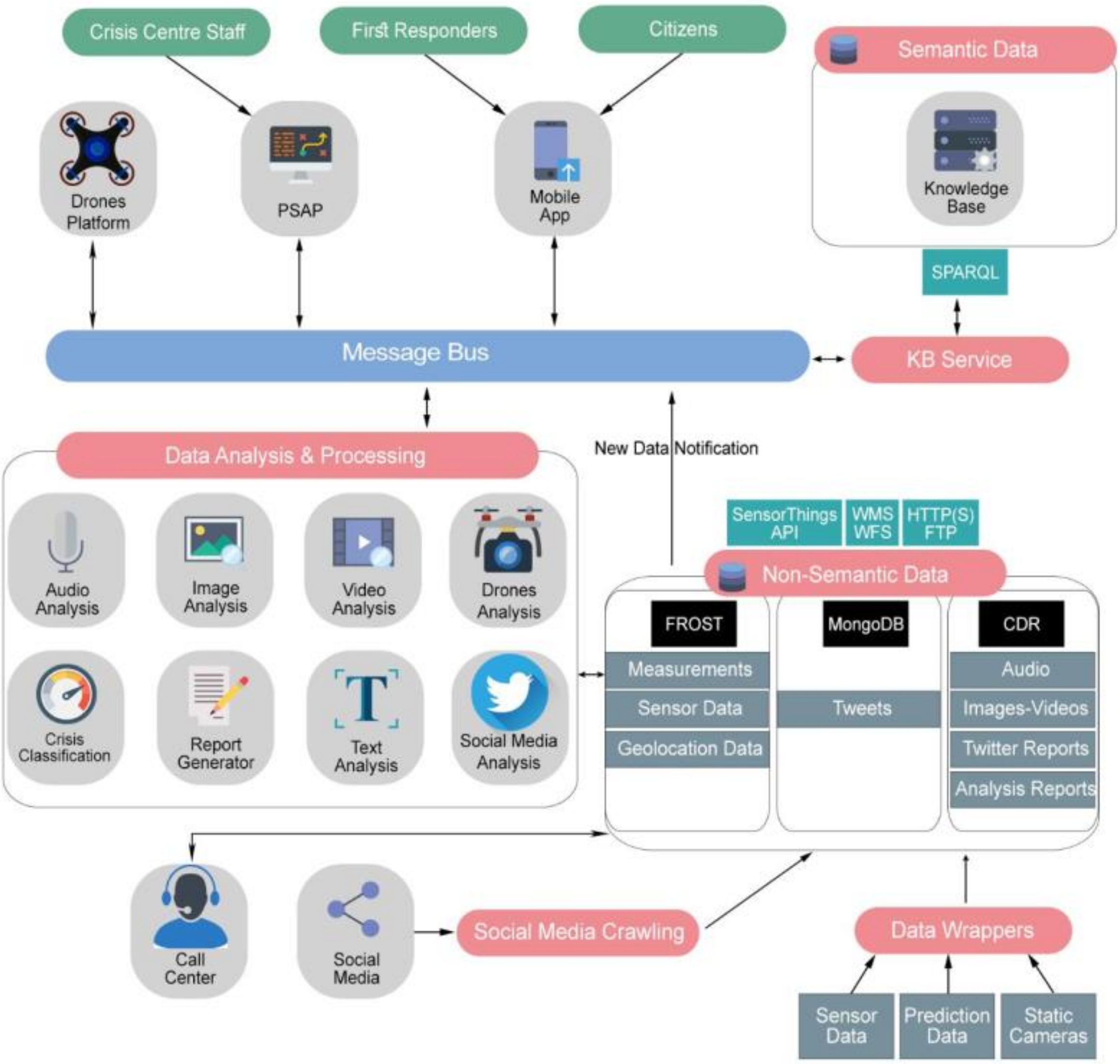



Figure 2: Architectue of the beAWARE system

## 1.1 Objectives

The objectives of T3.1, T3.2 and T3.3 for the milestone MS4 of WP3 were the following:

- Extend the range of contents extracted in all modules to cover the requirements of the 3rd pilot (fire scenario).
- Improve the computer vision deep learning methods employed to detect crisis events in visual content (images and videos).
- Improve the Automatic Speech Recognition (ASR) statistical methods to support messages in Spanish and English pertinent to the 3rd pilot.
- Extend the range of information extracted from multilingual textual inputs with information about possible states associated with events.
- Detect mentions to places and map them to specific geolocations in the text analysis module.

- Improve the text analysis methods to extract concepts and conceptual relations from texts in Spanish and English pertinent to the 3rd pilot.
- Add new functionality to support the operation of drones for surveillance tasks and the analysis of footage received from drones.
- Extend the multimodal fusion techniques for crisis classification in order to incorporate new contents produced by the analysis modules as well as the new drones analysis component.

## 1.2 Results towards the foreseen objectives of beAWARE project

The successful run of the 3rd pilot constitutes solid evidence that the main goals were accomplished. While more user-oriented evaluations will be presented in other deliverables, this document reports multiple technical evaluation that assess separately the performance of the individual components that makes the analysis and crisis classification modules. The results of each evaluation are discussed in the corresponding subsections of Section 7, and a summary of all results is provided in the conclussions in Section 8.

The final versions of the analysis modules can detect incidents in the multiple types of inputs foreseen: audio, text, images and video. In the case of audio and text, two languages are supported for each scenario and the text analysis module does not only detect incidents but also related information about impacted objects, locations and the state of an incident, e.g. if it is factual or hypothetical, if it is growing or decreasing in magnitude, etc. Detecting states associated with incidents was not part of the planned improvements, but it contributes greatly towars the production of more informative reports in WP5. It allows to report not only incidents like Fire, Wind or Heat but also wether they are happening or there is just a risk, how they are progressing and so on.

All the contents distilled by the analysis modules are projected to ontological representations and therefore can be easily integrated into the same semantic representation by the KBS (WP4). In addition, the crisis classification component along with the validator mechanism of KBS ensure that the PSAP receives only relevant incidents. The new geolocation functionality in the text analysis module is a key contribution towards the successful geospatial clustering of textual, social media and audio data.

Finally, the addition of input from drones and fixed cameras to the beAWARE platform is also a significant result reported in this document, following new user requirements and feedback from reviewers after the 2nd pilot.

## 1.3 Outline

This deliverable is structured as follows. Following the introduction, sections 2, 3, 4 and 5 describe the final versions of the analysis of image, video, drone footage, audio and text, as well as the advanced methods underpinning these modules. Section 6 reports the final version and advanced methods for crisis classification and risk assessment. Section 7 contains the evaluations and results for all components involved in the WP3 modules. Finally, section 8 contains an overall summary for the evaluation results and draws conclussions.

# 2 Image and video analysis

## 2.1 Overview

The task of visual analysis in the beAWARE project refers to concept extraction from multimedia content (images/videos), and it is supported by the IMAGE ANALYSIS and VIDEO ANALYSIS components. Through the project's lifetime the list of extracted visual concepts was gradually enriched in order to prepare the system for each added use case and the pilots that concluded each prototype's cycle of development and evaluation. This process involved the development of several computer vision and machine learning algorithms for an initial version of the system (i.e. first prototype) and later the refinement of those methods and the deployment of new ones, in order to meet additional requirements (i.e. for the second and final prototype). So far, we have covered all the visual analysis techniques that appear on the first prototype and even some regarding the second prototype in deliverable D3.3. In the following we provide details about the progress that has been made from the first prototype to the final version of the various modules.

## 2.2 Improvements of the modules in the final prototype

### 2.2.1 Emergency event detection

In order to recognize emergency events from images and videos the first version of the **Emergency Classification (EmC)** module was developed. The module was made to deal with the task of image recognition using a deep convolutional neural network (CNN) in order to classify each image into three possible classes: "flood", "fire" or "other". Note that it was possible to detect "fire" images even though the fire use case wasn't tested on the 1st or the 2nd pilot. When dealing with videos an entirely different approach was followed, where a dynamic texture recognition technique was developed in order to classify videos based on spatiotemporal low-level texture features (Local Binary Patterns, LBP). Unsupervised clustering (Gaussian Mixture Models, GMM) was then performed to construct a visual vocabulary with the most discriminant low level features. Finally, the Fisher Vector representation was adopted, transforming initial LBP-flow vectors of each video sample into a mid-level single vector representation, based on the detected most discriminating features (GMM vocabulary) of a training video database.

The final version of beAWARE system is expected to deal with smoke reports and as such the EmC module was upgraded for the final version with the addition of smoke recognition inside images and videos. Unfortunately, deep CNNs (like the one that the previous EmC version adopted for image analysis), which are trained to solve a N-class problem cannot be repurposed to solve a (N+1)-class problem without retraining from scratch. Therefore, a new data collection effort was required, which consists of gathering annotated instances of smoke images and adding them to the previous version's training set. Moreover, when dealing with dynamic texture recognition for videos, a new representation would have to be built for the texture of smoke. Smoke is a very delicate texture that can look a lot like fog or clouds and it can even be falsely detected when noise is present in an image due to its relatively simple colour range, which does not extend far beyond grey. Another drawback of the dynamic texture approach was that it added computational complexity to the EmC module, given that it is an entirely different framework compared to the deep CNN inference.

For those reasons we have made two critical changes to the EmC module:

1. We have completed a new cycle of public data collection in addition with the previously gathered beAWARE data we have managed to compose an extended set of over 20000 flood, fire and smoke images in order to fine-tune the final version of our deep CNN emergency classifier. It is common in the case of deep CNNs that more training data usually result to better performance.
2. A unified framework has been deployed to handle both images and videos, sharing a common function. More specifically, we have wired the newly trained CNN classifier to analyse images as well as sequential video frames, so as to achieve flood, fire and smoke detection in both images and videos using the same classifier without the need to deploy the dynamic texture recognition pipeline. An emergency label is now inferred for every video frame and a majority vote scheme is conducted in order to decide the emergency class of the full video. This way, the classifier serves simultaneously both components and increases the efficiency of visual analysis.

We have merged our previous set with the newly gathered data and we present the full collection in Table 1. In general, flood images have been taken from flooded areas like city streets and overflown rivers. Fire and smoke images in those datasets have been captured during forest wildfires, explosions, fires in urban environment and fires caused by riots. Fire images may contain flame and smoke textures, but the presence of flame is enough to characterize a ‘fire’ instance and smoke must appear alone without any flame in order to characterize a ‘smoke’ instance. The ‘other’ category serves as the negative class and is given to instances that do not show any of the three emergency events. Note that for every category different negative (‘other’) samples are provided, as non-flood, non-fire, non-smoke, since the deep CNN features that are going to be extracted need to discriminate between the important visual clues that uniquely characterize flood, fire and smoke textures and non-flood, non-fire, non-smoke instances. For example, the non-fire class contains sunset images. This is done in order to “instruct” the model to avoid learning colour as an important visual trigger for fire detection and give weight to texture features instead.

Table 1: Gathered datasets for flood, fire, smoke detection

| Dataset name | Flood images | Fire Images | Smoke Images | Other Images |
|---|---|---|---|---|
| Bowfire[1] | | 118 | | 107 |
| Corsican fire database[2] | | 1135 | | |
| MediaEval 2017 Multimedia Satellite Task[3] | 1920 | | | 3360 |

[1] https://bitbucket.org/gbdi/bowfire-dataset/src/master/

[2] http://cfdb.univ-corse.fr/index.php?newlang=english&menu=1

[3] http://www.multimediaeval.org/mediaeval2017/multimediasatellite/

| Dataset name | Flood images | Fire Images | Smoke Images | Other Images |
|---|---|---|---|---|
| European Flood 2013 Dataset[4] | 3151 | | | |
| Fire-Detection-Image-Dataset[5] | | 109 | | 537 |
| Fire-Smoke-Dataset[6] | | 1000 | 1000 | 999 |
| ForestFire[7] | | 223 | 45 | 420 |
| FiSmo (part of)[8] | | 1885 | 369 | 3583 |
| beAWARE data | 380 | 25 | 18 | 262 |

### 2.2.2 Segmentation of water and flame textures

Delving deeper into thorough analysis of the hazardous environments that are captured inside multimedia, the first version of the **Emergency Localization (EmL)** module was developed. This module was responsible to localize flood or fire regions inside images and videos. A deep CNN technique for semantic image segmentation was adopted, fine-tuning the network by using water and fire image regions from publicly available datasets, so as to characterize the pixels of images as “water”, “flame”, or “background”. The resulting “water” and “flame” pixels would then form semantically localized areas inside an image where the corresponding emergency event had a direct impact. In a similar fashion as the EmC module, a technique based on dynamic texture representation was developed in order to perform water and flame segmentation in video frames. The aforementioned LBP low-level descriptor was used for feature extraction. Superpixel clustering was carried out in a multi-layer scheme creating a final descriptor, which characterizes the areas of the video frames. Finally, the discriminative models, that have been trained for the EmC video classification task, were used in order to localize the desired dynamic texture in a spatio-temporal manner. The decision was conducted locally for each area covered from superpixels of the top layer.

Moving onto the final version of this module we decided to adopt the same framework as we did for the final version of the EmC. Specifically, we perform image segmentation using the aforementioned EmL deep CNN for every video frame instead of deploying the full dynamic texture pipeline. As with the case of EmC, this allows us to repurpose our models and use them in an efficient manner.

[4] https://github.com/cvjena/eu-flood-dataset

[5] https://github.com/cair/Fire-Detection-Image-Dataset

[6] https://github.com/DeepQuestAI/Fire-Smoke-Dataset

[7] https://dataturks.com/projects/qdyzl1013/Forest%20Fire

[8] https://github.com/mtcazzolato/dsw2017

### 2.2.3 Detection of people and vehicles

In addition to the analysis of the images and videos regarding the actual disaster conditions, identifying the impacted targets, such as pedestrians and vehicles, is an important function for the beAWARE system. The task of human and vehicle detection in images and videos was carried out by the **Object Detection (ObD)** module. This module was responsible to provide a set of bounding boxes of the persons and vehicles in images and video frames, as well as their immediate surroundings. Groups of people and individuals were catecorized as “person”, while vehicles would result to one of the following categories: “car”, “truck”, “bus”, “bicycle” or “motorcycle”. Object Detection works on two stages. At stage 1 the task of **detection** is carried out in order to provide candidate bounding boxes that may contain the individuals or the vehicles. At stage 2 the task of **recognition** is carried out in order to classify the boxes based on their visual content. For the detection stage we adopted a deep CNN feature extractor in order to provide appearance features for a number of bounding box candidates and then a Region of Interest (RoI) pooling scheme was used for the recognition stage so as to refine bounding box coordinates and finally classify their content.

This task is more complicated when dealing with videos. Several pedestrians or vehicles may appear in multiple sequential video frames and then disappear completely or reappear again at a later time as a result of their individual motion or the motion of the camera. Simply detecting all the instances at each video frame would be redundant and inaccurate as the same targets would appear in multiple frames. As a result, visual tracking of the detected bounding boxes was incorporated in the ObD module for videos. The tracker accepts new image patches as input queries (the detected bounding boxes) and is assigned to discover the most probable position of each query in subsequent frames using kernelized correlation filters to match visual appearance.

Each concept detected from the visual analysis components should be meaningful for the rest of the system to be processed and finally reported to the authorities. Different emergency circumstances may have varying impact across individuals with or without disabilities and other living beings, such as animals. Thus, we aim to provide a better support in decision making by recognizing as many concepts as possible. Therefore, in an effort to provide more thorough visual analysis reports to the beAWARE system we extended the ObD functionality to include the detection of animals and wheelchair users separately from the generic “person” category. In order to do so we adopted the Faster-RCNN (Huang et al., 2017) architecture with the Inception-Resnet-v2 module to serve as the feature extractor of stage 1. We acquired the pre-trained weights of this model on the Open Images v4 (Kuznetsova, et al., 2018) dataset that includes instances from all the previous concepts plus the new ones (wheelchair users, cats and dogs) and deployed it in a similar fashion as the previous version.

### 2.2.4 Inference of severity levels

On top of the main modules (EmC, EmL, ObD) an initial version of the severity level estimation feature was developed whose purpose was to fuse the main modules outputs so as to infer a measure of severity for each bounding box detected by the ObD. This is done in a qualitative manner using three levels: ‘Low’, ‘Medium’, ‘High’, which can also be interpreted as a three-level qualitative risk assessment scale: 'Safe target', 'Target possibly in danger', or 'Target in danger' respectively.

The possible outcomes were derived in the first version as follows:

a) ‘Low’: EmC classified the candidate image or video as 'other'. All the detected bounding boxes from the ObD were declared 'Safe targets'.
b) ‘Medium’: EmC detected ‘fire’ or ‘flood’ image or video. All targets that were detected from ObD were automatically characterized as 'Possibly in danger'.
c) ‘High’: A target with a ‘Medium’ severity level when its bounding box overlaps with EmL emergency masks (i.e. ‘fire’ or ‘flood’ regions in the image/video).

We have since realized that it would be preferable to provide qualitative levels only for the overall situation in an image or video and construct quantitative severity indicators instead for each target that appears inside.

The severity indicator that is calculated for each target, which we call risk factor, is a numerical value that represents the overall percentage of overlap between a target’s bounding box (taken from ObD) and a fire or flood region (taken from EmL) and can take values in the range [0,1].

For the severity of an individual image or video the possible outcomes are now defined as:

a) ‘Unknown’: When ObD has detected some targets, but neither EmC or EmL have detected possible emergency regions in the image/video.
b) ‘Severe’: If EmC detects flood, fire or smoke but not any living target or vehicle has been detected from ObD.
c) ‘Extreme’: If from both EmC and Obd an emergency situation has been detected that may possibly have an impact to living targets or vehicles.

Besides solving an important compatibility issue with the rest of the system, this update also adds flexibility to the system because it enables incidents created by the visual analysis results to be later modified in order to reflect various aspects of the current situation that visual analysis is agnostic to (e.g. incident clustering).

### 2.2.5 Validating multimedia content related to beAWARE

The validity of content that is circulated in the beAWARE system is checked through a two-layer validation scheme. In the first layer three components, the image and video analysis and the social media analysis, enclose mechanisms to act as a first barrier, filtering out malicious or false information. The complete scheme is described in detail in D4.3.

Image and Video Analysis are responsible to examine every image/video item that is forwarded to the system through the Mobile App. With the exception of some operations defined with the help of exception flags, the components always filter the results through the internal validation mechanism. The exception flags are:

- Accreditation documents are forwarded through the Visual Analysis system without examination, using the appropriate flag in the Mobile App’s report form.
- Other items sent from inside the beAWARE system are not filtered, since they are considered already validated. One such case was the traffic analysis on the Angeli Bridge in Vicenza during the 2nd pilot. These videos were sent from the Visual River Sensing component, in order to further analyze the traffic on top of the bridge whenever the water underneath was found to exceed a predefined level.

Every request for visual analysis that doesn't carry any of the above exception flags is analyzed and the results filtered through the internal validation mechanism.

The validation mechanism was first developed and integrated for the 2nd pilot, where the flood use case was under examination. Therefore, in the first version each image or video that was found by the Emergency Classification module (EmC) as flood-related was forwarded through the filter. If an image or video didn't contain any flooded areas the filter discarded the request and no further analysis was performed or results forwarded to the system. With the addition of smoke and fire incidents during the 3rd pilot, it is not correct to assume anymore that an image or video that contains targets but not actual flames or smoke is not relevant to the system, since it is considered dangerous for a person, a vehicle or an animal to be located not only on the actual burning site but even at a nearby location. Consequently, the validation mechanism has been updated for the final prototype in order to forward items that show either an emergency taking place, or targets that are possibly in danger (i.e. people, animals and vehicles) even if no flames or smoke is detected.

## 2.3 Visual River Sensing

As described in deliverables D7.5 and D7.6, in the second version of beAWARE platform, additional functionalities have been added to the visual analysis component, in order to analyze footage from static cameras, for remote river monitoring. The new module is called Visual River Sensing (VRS) and performs visual analysis on videos from static surveillance cameras installed next to the river, in order to estimate the water level and to generate an alert, when a threshold is exceeded. For the needs of the flood pilot, two static surveillance cameras were installed in the wider area of Vicenza, which were used for flood monitoring and are connected to the rest of the platform through IP streaming. VRS was calibrated for the surveillance camera installed in Bacchiglione river in the center of Vicenza (Angeli Bridge) and can easily be adjusted to other cameras. VRS streams video-frames directly from the IP of the camera and creates a short video chunk in order to be processed. An example of a captured frame can be seen in Figure 3. The figure depicts Angeli bridge, a part of the Bacchiglione river and an old rod (marked inside a red box), which is placed on the bank of the river and was used for measuring the water level, before the installation of water level sensors. Apart from water level estimation, the video chunk is also forwarded to the ObD module for traffic estimation, in order to obtain a better overview of the flood event.

Figure 3: Captured frame for the static camera in Bacchiglione river (Angeli Bridge). The marker has been marked with a red box

The water level estimation module uses an edge detection algorithm in order to detect the marker (rod), which is of known length. After detecting the marker, the algorithm estimates its length in pixels (the visual part of the marker which is above the surface of the water) and corresponds it to its real length in meters, by using calibration data. Subsequently, the length of the visible part is corresponded to water level of the river. If the water level exceeds some predefined thresholds, three different types of alerts are generated respectively: 'Moderate', 'Severe ', 'Extreme'. In the second version of the platform, VRS was using only the first frame of the video for the water level estimation. In order to become more robust, the final algorithm has been modified and uses average values from multiple frames. Figure 4 visualizes the result of the water level estimation algorithm. The picture on the left depicts a part of the captured frame of Figure 3 which is used by the edge detection algorithm, in order to extract the rod's boundary. The picture on the right depicts the result of the edge detection algorithm.

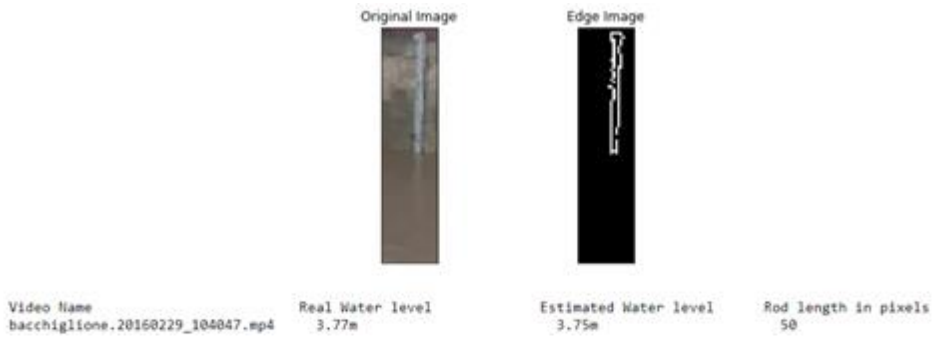



Figure 4: An example of the edge detection algorithm: Left image contains the cropped image of the rod and right image the corresponding edge detection result

# 3 Drone Analysis

## 3.1 Drone as a content provider

Drones are being used for commercial activities these days, mostly requiring one pilot per drone to operate it. On board equipment stores information locally which needs to be downloaded and transmitted upon landing. Some autonomous flying capabilities do exist, but the route is pre-programmed before the drone takeoff.

The drones platform consists of 3 components:

1. Drones server – On which the different service types are deployed, and specific service instances are run. Creates generic execution blocks for the drone based on the service and its specific configuration. It provides a drone vendor/model independent SDK for developers to create new service types.
2. Drones edge device – Receives instructions from the server component and interacts with the drone via a specific SDK to fulfill the service. It received the captured data from the drone. In addition, it enables the configuration and control of a specific service instance.
3. Platform dashboard – provides a real-time view of the service as its being carried out, including the current location of the drone, the following locations to be visited and the path taken. In addition, it displays relevant analysis results and current state of the service.

The developed drones platform provides the following capabilities:

- Autonomous piloting – programmatic piloting without the need of a drones pilot. Generic commands are created to support agnostic control of drones; translated to a specific SDK based on the drone currently in action. The drone is controlled programmatically.
  Helps to remove unnecessary attention during a crisis and to focus on flight analysis and outcome rather than the flight operation itself.
- Route planning – calculate flight parameters for optimal coverage of the area to be scanned. Taking into consideration overall area, maximum flight time, height, and desired overlap (to ensure complete coverage of the area to be scanned).
- Dynamic route planning – Route is not fully loaded to the drone upon takeoff, but is rather created one step at a time, in real-time, while the drone is in the air. This enables incorporating analysis results in determining the continuation of the route. This capability makes the drones platform extremely useful as a general inspection can be performed, and once a specific incident of interest is spotted, the drone can be directed to take a closer look without the need to start a different mission.
- Control of on-board equipment – equipment such as cameras and possibly additional sensors can be controlled during the flight.
- real-time transmission of acquired data – data produced and accumulated by on board equipment can be transmitted in real-time, for example to the analysis module and to the drones control dashboard.

Within beAWARE the drones platform acts as an additional content provider with a unique point of view which is able to provide content fast, in a safe manner and even in areas which are difficult to reach by humans and ground vehicles.

The drones platform establishes a bi-directional communication path with the beAWARE platform, based on the established collaboration tools in the platform. The drones platform captures media from on board equipment and passes it to the beAWARE platform. At this stage the data consists of pictures and videos. The data is first uploaded to the cloud-based object store, and once the data has been stored a message is sent over the cloud-based message bus to alert all interested components that a new drone-based information is available.

Data and control flow of the drones platform:

1. A service type is selected and a specific service instance is configured and launched using the edge device.
2. The server part executes the service instance and produces the next execution block.
3. The edge device receives the execution block, which is drone agnostic, instantiates it for the specific drone in question and passes the new commands to the drone via its specific SDK.
4. Drone events and media are transmitted to the edge device, and from there they are passed to the server component.
5. The server component stores media in the object store and publishes an appropriate message on a specific topic of the message bus
6. Media and analysis results are pushed by the server to the drones dashboard.

The drones platform uses a topic called TOP031_UAVP_MESSAGE, which includes information about the location, altitude, heading, and gimble pitch of the drone. In addition, it includes the path from which the data can be retrieved. Figure 5 shows the path in which drones collected information flows through the different system components. The drones platform publishes a message describing a new piece of data being available. The Drones analysis module subscribes to notifications from this topic, and thus picks up the new data. When the analysis is done the drones analysis module publishes the results, which are picked up by the knowledge base, which in turn asks for a report to be generated before sending a message to which the PSAP subscribes.

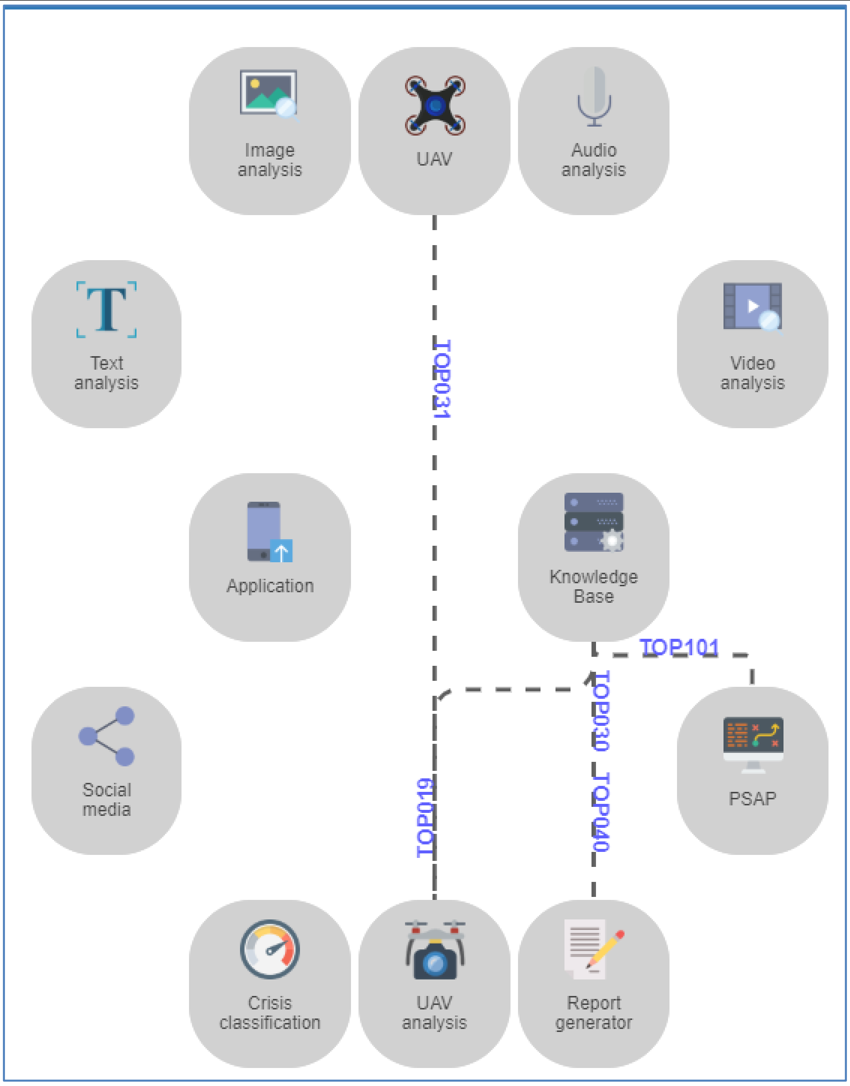



Figure 5: Drones data processing path

The drones platform in turn also registers as a message consumer to obtain analysis results from the drones analysis module (TOP019_UAV_media_analyzed detailed in Table 2). That, along with the dynamic routing feature, enables the drone platform to set in real-time a new route to be followed by the drone, based on the analysis results. The message from the drones analysis module contains a link to the original media file, a link to the analyzed media file, and a link to the analysis results.

The drone can pick up this information. For example, in one of the scenarios the drone scanned an area which included a “person in danger”, analysis on the drone imagery was performed while the drone completed scanning the area. At that time the drones platform picked up the message from the drones analysis module, which identified the location of a person-in-danger, and a new execution block was created for the drone to fly to the identified location to take (and broadcast) a closer look, before returning to its landing place.

Table 2: Media analyzed message format

| Field | Type | Description | Mandatory (Y/N) |
|---|---|---|---|
| media_original | String | Link to the original media file | Y |
| media_analyzed | String | Link to the analyzed media file | Y |
| media_analysis | String | Link to the JSON file that was produced from the analysis | Y |
| media_timestamp | Datetime | The timestamp of the media creation | Y |
| location | | The location of the media creation (UAV), defined with latitude and longitude | Y |
| latitude | Decimal | The geographic latitude of the media | Y |
| longitude | Decimal | The geographic longitude of the media | Y |
| incidentID | String | The incidentID as received from the UAVP in topic 031 | Y |

## 3.2 Drone Analysis Module

As already described in deliverables D7.5 and D7.6, at the second version of the beAWARE platform a new analysis module was added in order to take advantage of the integration of drones in the beAWARE platform. The new module is called **Drones Analysis (DA)** and is based on deep-learning object-detection techniques and models trained by CERTH on drone footage. DA is responsible for receiving drone footage, perform visual analysis in order to detect people and vehicles in danger and detect disasters (flood/smoke/fire) and subsequently inform the Drones Platform and PSAP about the results. At the second version of the platform, the module was able to handle sequences of images, sent at a frame rate of 1 fps. DA was creating batches of consecutive individual images by setting a predefined time threshold for the creation of each batch (10 seconds). Subsequently, each batch was individually analyzed, by performing object detection and tracking, in order to detect people and vehicles in danger, along with image classification in order to detect disasters in images. In case of a positive detection, an alert was created (through dedicated bus messages), containing the type of objects, their position and the analyzed sequence as a video. The results were communicated to the PSAP and the Drones Platform. Specifically, the drones platform was able to navigate the drone back to the target, by using location information. Apart from the full detection model (containing people and vehicles) that was presented in deliverable D1.3, for demonstration purposes, a separate model was trained during the second pilot, able to detect a dummy that was used to simulate a person in danger. Since the dummy does not exactly resemble a human being, in order to avoid misclassifications or adaptation of the existing model to the dummy, a separate model had to be trained on footage of the dummy. Evaluation of this separate model was presented on D7.6.

At the final version of the platform, the DA has been extended in order to be able to handle both image and video sequences. In case the input is an image, the DA creates a batch and waits until the maximum number of images is reached before proceeding to analysis. A single output message is generated for the whole batch. In case the DA receives a video chunk, it subjects it directly to analysis and creates an output message for each video chunk. Several modifications had to be applied in order to adjust the module to video input. First of all, topic messages used for communication between the Drones Platform and the DA were modified in order to contain multi-frame metadata instead of single-frame metadata that were used in image sequences. Additionally, in order to be able to analyze high frame rate videos without degrading performance, input videos along with corresponding metadata are being sub-sampled into a predefined frame rate (maximum 5fps).

Moreover, during the final version, a new task was added in order to facilitate and monitor evacuation processes during crises. Specifically, when a drone is sent on an evacuation mission, footage from the area under evacuation is analysed in order to detect if any person is left behind. Alerts are created every time a person is detected on an evacuated area during the mission. At the end of the mission, the drone sends a notification that the mission is over and a relevant report is created for the whole drone mission. In order to distinguish the different DA functionalities, three distinguished analysis tasks have been defined, which can be selected separately or in combination, according to an input variable. The first task is ‘Object Detection’ during which the DA performs object detection and tracking in order to detect people and vehicles. The second task is ‘Crisis Detection’ during which the DA performs image classification in order to detect disasters. The model used for the classification is the same that is used by the EmC module, that was described in section 2.2.1. Finally, the third analysis task is ‘Evacuation’ during which DA detects only people and generates relevant evacuation reports. Several post processing steps were added in order to improve tracking and eliminate false positives, such as scattered detections of objects or disasters.

# 4 Audio analysis

The **Automatic Speech Recognition (ASR)** component is used in combination with **Multilingual Text Analyser (MTA)** in order to automatically extract information from emergency calls and audio messages. The purpose of ASR is to receive audio recordings, either through the Mobile App or as emergency calls to a dedicated call center and provide transcriptions, which are forwarded to MTA for semantic extraction. Until the second version of the platform, the Italian and Greek acoustic models had been adapted to case-specific recorded speech, in order to enhance emergency-related terminology. Additionally, the corresponding dictionaries had been cleared from erroneous or rare words. At the second version of the platform, a call-center solution was also included in the platform, in order to receive emergency phone calls, and a relevant module was developed, able to fetch recorded calls and forward them to ASR. During the call, the caller is able to determine his/her language, through an **Interactive Voice Response (IVR)**, in order for the call to be forwarded to the corresponding ASR language model. Apart from the integrated call center, for the needs of the third pilot and especially for integrating legacy tools with the beAWARE platform (blended phase), the PLV call center was used in order to record emergency calls and store them on an ftps server. A dedicated service was set to check for new audio files and forward them to beAWARE platform.

During the third development period, the focus was mainly on the Spanish model, which was also the official language of the third pilot. In order to add missings words and location names in the Spanish model and to enhance emergency-related terminology, the Spanish dictionary and language model had to be adapted. A set of 95 Spanish phrases was extracted from the tweets list that were formed by PLV. The Spanish dictionary was updated with missing words and the language model (which contains occurrence probabilities of single words and combinations of two or three words in a language corpus) was updated with new occurrence probabilities from the formulated set of phrases. Apart from model adaptation, some extra modifications were made during the third development phase in order to improve recognition accuracy. One modification refers to the format of the language model. In order to be able to use language models of unlimited vocabulary size with improved memory consumption and query speed, CMU Sphinx provides a function (sphinx_lm_convert) in order to convert language models from text format to trie binary format[9]. At the second version of the platform the Spanish language model was converted to trie binary format, mainly because of the size of the original model. However, after comparison between different formats, a small deterioration of the accuracy of the binary format was noted, as described in Section **Error! Reference source not found.**, which is probably caused by a defect of the convertion function. At the final version of the component, the language model was reverted back to its original format, improving this way the accuracy, without significant increase in analysis time. Additionally, there was a fix on the quality of the audio converter that was used on the Mobile App to handle audio files. Initially, received audio files were converted to low quality format, which caused some loss of information. At the last version this issue was solved and audio files are converted directly to the high quality .wav format (BitRate = 256 kb/s). Thus, speech recognition accuracy was improved for the audio coming from the Mobile App after improving the audio quality. However, speech recognition accuracy on audio coming from both the

[9] https://cmusphinx.github.io/2015/07/new-language-model-binary-format/

integrated call center solution and the PLV call center is still affected by a low audio quality, because of the default recording parameters (recording bit rate around 16kb/s). These parameters depend on third parties and cannot be modified.

# 5 Text analysis

The **Text Analysis (TA)** component produces an ontology-ready knowledge representation using a combination of NLP and IE tools. As reported in D3.3, the basic version of the text analysis component performed a linguistic analysis on the input sentences and then mapped the results to a proposed knowledge representation. In contrast to the linguistic analysis tools already introduced in D3.3, which can analyse any text received by the TA, the basic mechanism, used to produce the final knowledge representation in the 1st pilot, was limited to a restricted set of inputs. Since then, our efforts have been directed towards obtaining a wide coverage knowledge extraction strategy, that is not constrained to materials scripted for the project pilots and that is also capable of extracting useful information even from noisy audio transciptions and ungramatical social media texts.

In the following we describe the final version of the text analysis component of the beAWARE system. We start by reporting the updates to the linguistic analysis component and its instantiation for each of the project languages - English, Greek, Italian and Spanish. Then we describe the knowledge extraction procedure, presenting the final version of knowledge respresentation passed on to the beAWARE system, and detailing the components used to obtain it.

## 5.1 Linguistic analysis

The final version of our linguistic analysis component uses the UD-based analysis pipeline described in D3.3 (Sections 5.1.1 to 5.1.3), which produces both standard UD syntactic parses and a deeper representation based on UD that we refer to as UD graphs. This latter representation aims at removing language-specific constructions and linguistic information to facilitate knowledge extraction. We chose to use a UD-based linguistic analysis component over the alternative pipelines described in D3.3 that use language-specific syntactic annotation schemes. This choice was motivated by the poorer performance of the language-specific pipelines. In addition, the UD component was easier to apply to multiple languages due to its common annotation scheme.

The final UD linguistic analysis component comprises the following tools:

- An off-the-shelf trained statistical parser that produces UD syntactic trees
- UPF's graph-transduction grammars to convert the shallow syntactic structures into deep syntactic structures, or UD graphs

UPF's efforts since D3.3 on the relation extraction pipeline focused on the following key aspects:

- update the grammars to reflect the latest changes in the original UD annotations (v2.4),
- update the grammars to produce the UD graphs used in the final version of the knowledge extraction component,
- demonstrate the usefulness of the deep structures by applying them to the automatic creation of multilingual datasets used as training material for research challenges,

- and test the portability and reusability of the graph-transduction grammars beyond the scope of beAWARE, through the adaptation to other languages not foreseen in the project.

In the following we will describe the linguistic structures and the grammars used to obtain one representation from the other.

### 5.1.1 General specification of the linguistic structures

The analysis produced by the off-the-shelf UD parser includes dependency-based syntactic trees annotated with lemmas, part-of-speech tags, morphological and dependency information under the form of grammatical functions such as *subject*, *object*, *adverbial*, etc. We have developed a grammar-based deterministic grammar to convert UD trees to a deeper more abstract representation closer to linguistic meaning and more suited to knowledge extraction. This converter produces three types of structures:

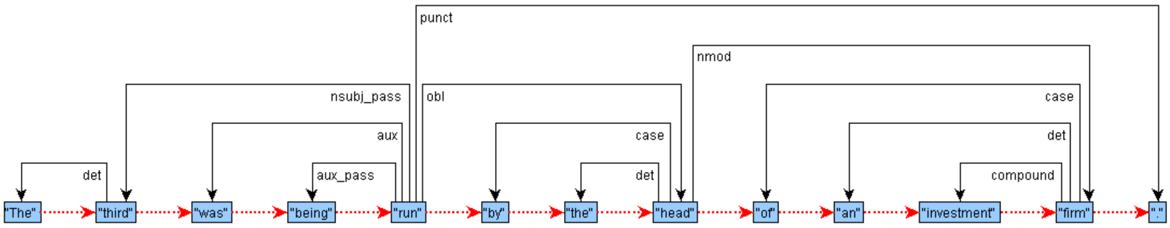
1. Shallow structures: unordered UD dependency trees with lemmatised words that hold PoS tags and morphological information
    - Languages for which we have annotated corpora: Arabic, Chinese, Czech, Dutch, **English**, Finnish, French, **Greek**, Hindi, Indonesian, **Italian**, Japanese, Korean, Portuguese, Russian, **Spanish**
2. Deep structures: unordered predicate-argument tree with lemmatized content words that hold coarse-grained PoS tags and semantic information
    - Languages for which we have annotated corpora: Chinese, **English**, French, **Greek**, **Italian**, Portuguese, **Spanish**
3. UD graphs: unordered graph resulting from applying certain transformations of the deep structures, e.g. predicative words govern their arguments in all cases.

Shallow structures are obtained by simply removing the order and surface forms information from the original UD parses produced by a trained statistical parser. Other than this, they are identical in all respects to the UD specification.

Deep structures consist of predicate-argument structures obtained through the application of graph-transduction grammars to the UD surface-syntactic structures. The deep and surface structures are aligned node to node. In the deep structures, we aim at removing all the information that is language-specific and oriented towards syntax:

- determiners and auxiliaries are replaced (when needed) by attribute/value pairs, as, e.g., Definiteness, Aspect, and Mood:
    - auxiliaries: *was built-> build*;
    - determiners: the building-> building;
- functional prepositions and conjunctions that can be inferred from other lexical units or from the syntactic structure are removed:
    - built by X-> built X
- edge labels are generalised into predicate argument (semantics-oriented) labels in the PropBank/NomBank fashion:
    - subject(built, by X)-> FirstArgument(build, X)

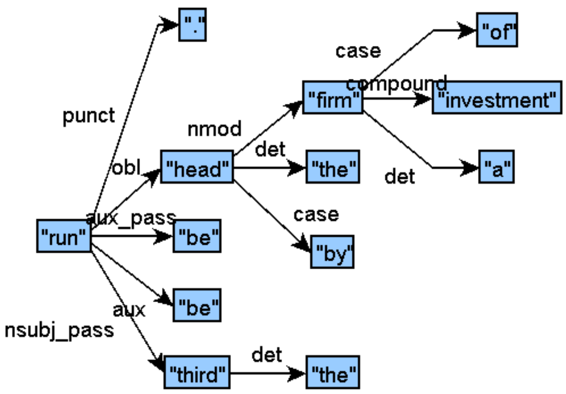
Figure 6, Figure 7, and Figure 8 show original, surface and deep structures respectively.




```
# sent_id = weblog-juancole.com_juancole_20051126063000_ENG_20051126_063000-0006
# text = The third was being run by the head of an investment firm.
```

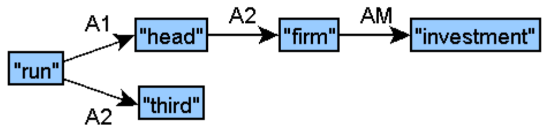
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | The | the | DET | DT | Definite=Def\|PronType=Art | 2 | det |
| 2 | third | third | ADJ | JJ | Degree=Pos\|NumType=Ord | 5 | nsubj_pass |
| 3 | was | be | AUX | VBD | Mood=Ind\|Number=Sing\|Person=3\|Tense=Past\|VerbForm=Fin | 5 | aux |
| 4 | being | be | AUX | VBG | VerbForm=Ger | 5 | aux_pass |
| 5 | run | run | VERB | VBN | Tense=Past\|VerbForm=Part\|Voice=Pass | 0 | root |
| 6 | by | by | ADP | IN | _ | 8 | case |
| 7 | the | the | DET | DT | Definite=Def\|PronType=Art | 8 | det |
| 8 | head | head | NOUN | NN | Number=Sing | 5 | obl |
| 9 | of | of | ADP | IN | _ | 12 | case |
| 10 | an | a | DET | DT | Definite=Ind\|PronType=Art | 12 | det |
| 11 | investment | investment | NOUN | NN | Number=Sing | 12 | compound |
| 12 | firm | firm | NOUN | NN | Number=Sing | 8 | nmod |
| 13 | . | . | PUNCT | . | _ | 5 | punct |

Figure 6: Original UD structure in the CoNLL-U format (top: graphical representation)




| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | the | _ | DET | DT | Definite=Def\|PronType=Art | 2 | det |
| 2 | third | _ | ADJ | JJ | Degree=Pos\|NumType=Ord | 3 | nsubj_pass |
| 3 | run | _ | VERB | VBN | Tense=Past\|VerbForm=Part\|Voice=Pass | 0 | root |
| 4 | be | _ | AUX | VBD | Mood=Ind\|Number=Sing\|Person=3\|Tense=Past\|VerbForm=Fin | 3 | aux |
| 5 | be | _ | AUX | VBG | VerbForm=Ger | 3 | aux_pass |
| 6 | head | _ | NOUN | NN | Number=Sing | 3 | obl |
| 7 | . | _ | PUNCT | . | _ | 3 | punct |
| 8 | by | _ | ADP | IN | _ | 6 | case |
| 9 | the | _ | DET | DT | Definite=Def\|PronType=Art | 6 | det |
| 10 | firm | _ | NOUN | NN | Number=Sing | 6 | nmod |
| 11 | an | _ | DET | DT | Definite=Ind\|PronType=Art | 10 | det |
| 12 | investment | _ | NOUN | NN | Number=Sing | 10 | compound |
| 13 | of | _ | ADP | IN | _ | 10 | case |

Figure 7: Shallow track input in the CoNLL-U format (top: graphical representation)




| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | third | _ | ADJ | _ | Degree=Pos | 2 | A2 |
| 2 | run | _ | VERB | _ | Tense=Past\|Aspect=Progr | 0 | ROOT |
| 3 | head | _ | NOUN | _ | Number=Sing\|Definiteness=Def | 2 | A1 |
| 4 | firm | _ | NOUN | _ | Number=Sing\|Definiteness=Indef | 3 | A2 |
| 5 | investment | _ | NOUN | _ | Number=Sing | 4 | AM |

Figure 8: Deep track input in the CoNLL-U format (top: graphical representation)

### 5.1.2 A converter for obtaining the UD graphs

The beAWARE deep UD grammars we use to obtain UD graphs from UD syntactic trees do not use any lexical resources; the predicate-argument relations are derived using syntactic cues only. The deep input is a compromise between (i) correctness and (ii) adequacy in a generation setup. Indeed, the conversion of the UD structures into predicate-argument structures depends not only on the mapping process, but also on the availability of the information in the original annotation.

As described in D3.3, our Deep UD grammars are rules that apply to a subgraph of the input structure and produce a part of the output structure. During the application of the rules, both the input structure (covered by the left side of the rule) and the current state of the output structure at the moment of the application of a rule (i.e., the right side of the rule) are available as context. The output structure in one transduction is built incrementally: the rules are all evaluated; the ones that match a part of the input graph are applied and a first piece of the output graph is built; then the rules are evaluated again, this time with the right-side context as well, and another part of the output graph is built; and so on. The transduction is over when no rule is left that matches the combination of the left-side and the right-side.

Consider, for illustration, a sample rule from the SSynt-DSynt mapping in Figure 9. This rule, in which we can see the left-side and the right-side fields, collapses the functional prepositions (*?Xl*, identified during the pre-processing stage with the *BLOCK=YES* attribute/value pair) with their dependent (*?Yl*). That is a functional preposition such as *by* in *built by Y* is removed from the output structure and made to correspond with the right-side node *Y* (i.e., the dependent).[10] The right-side context is indicated by the prefix *rc:* before a variable or a correspondence. In practice, it means that the rule looks for the *rc:*-marked elements in the current state of the output structure, and builds the elements that are not *rc:*-marked. In this specific case this is the correspondence between the right-side *Y* and the left-side *by* and the new feature *original_deprel*, which stores the left-side incoming dependency relation. A similar rule would apply to *firm* and *of*, where *of* is the dependent in this configuration (see Figure 7). As a result of the application of this rule, only *firm* is left in Figure 8, which has a correspondence with both *firm* and *of* from Figure 7.

[10] Correspondences are meta-information used during the transduction; they are not explicit as such in the output structure. In order to maintain the alignments between surface and deep nodes, attribute/value pairs can be used: e.g. if *by* has a surface identifier "id=2", and *Y* id = "3", the deep *Y* node could have two identifiers "id=2,3" to mark the correspondence.

```
c:?Xl {                                   rc:?Yr {
  BLOCK = YES                               rc:<=> ?Yl
  c:deprel = ?dep                           <=> ?Xl
  c:id = ?i1                                original_deprel = ?dep
  c:?s-> c:?Yl {                          }
    c:id = ?i2
  }
}

(?s == PMOD | ?s == IM | ?s == SUB)
```

Figure 9: A sample graph-transduction rule; ? indicates a variable; ?Xl{} is a node, ?s-> is a relation, a=?b is an attribute/value pair

Table 3 sums up the current state of the graph-transduction grammars and rules for the mapping between surface-syntactic UD trees and UD graphs. The main improvements since D3.3 are (i) an increase of the number of rules, to account mainly for the coverage of the new languages and of new phenomena present in the latest version of the UD data, and (ii) the inclusion of an additional rule set for facilitating knowledge extraction in the context of beAWARE.

Table 3: Graph-transduction rules for UD-based deep parsing

| Grammars | #rules D3.3* | #rules D3.4* | Description |
|---|---|---|---|
| Pre-processing | 76 | 93 | Identify nodes to be removed<br>Identify verbal finiteness and tense |
| Shallow to deep UD | 120 | 147 | Remove idiosyncratic nodes<br>Establish correspondences with surface nodes<br>Predict predicate-argument dependency labels<br>Replace determiners, modality and aspect markers by attribute-value feature structures<br>Identify duplicated core dependency labels below one predicate |
| Post-processing | 60 | 73 | Replace duplicated argument relations by best educated guess<br>Identify remaining duplicated core dependency labels (for posterior debugging) |
| Deep UD to Graph UD | - | 70 | Converts the deep UD tree into a graph |

## 5.2 Knowledge extraction

The UD graphs produced by the linguistic analysis tools are not suited for consumption in knowledge-oriented tasks in the beAWARE system. The goal of the knowledge methods described in this section is to transform deep syntactic graphs into a unified knowledge representation based on concepts and conceptual relations rather than the words and linguistic relations that make up dependency-based representations.

While terms like *concept* and *conceptual relations* are vague and subject to multiple interpretations, we use them to refer exclusively to the ontological types and relations defined in the semantic representation resulting from T4.3. Thus, our goal is to produce a representation with instances of types present in the project ontology -*concepts*-, where these instances are interraled with relations also present in the ontology -*relations*.

First, we will describe the updated knowledge representation, and then we will detail the final version of the knowledge extraction pipeline used to obtain instances of this representation from texts.

### 5.2.1 Updated Knowledge Representation

Figure 10 shows an example of the output of TA in JSON format. Compared to similar examples in D3.3, the overall structure remains the same. The output contains one or more entitities, which instantiate ontology classes, as indicated in the *type* attribute. Entities that instantiate ontological subclasses of the class *Incident*, e.g. *Fire* or *Flood*, can have one or more participating entities. These participants may indicate either a vulnerable object impacted by the incident, a state of the incident, or a location in which the incident takes place. None of these entities can have further participants.

The beAWARE ontology comprises a taxonomy of types of vulnerable objects pertinent to the project use cases, e.g. vehicle or person. It also distinguishes between several types of states, i.e. *Risk*, *Greater*, *Lesser*, *Start* and *End*, and contains a class *Location* without any subclases. In addition to incidents, entities in the TA output may have any of these classes as their types. Participants adopt roles which correspond to object properties defined in the ontology, e.g. *participant*, *state* and *location*. Returning to the example in Figure 10, an event of type *Fire* has been detected that is impacting an entity of type *Forest* -a subtype of *Vulnerable Object*, *Asset* and *Ecological Asset*-, has a state of type *Risk* indicating its hypothetical status, and has a geolocated entity of type *Location* corresponding to the Valencian town of *El Saler*.

Since only incidents can have participants, the overall structure of the TA output corresponds to a forest of trees of depth one, the root of each tree indicating an incident and its children indicating impacted objects, states and locations. In this knowledge representation, each tree can also be interpreted as an n-ary relation indicating an occurrence of an incident and a set of entities participating in the event. This view is illustrated by the diagram in Figure 11.

Since the basic TA reported in D3.3, new concepts have been added to the ontology in response to pilot requeriments, and the TA has been duly updated to support them. Nevertheless, the general structure of knowledge representation as a set of event-centric n-ary relations has remained the same. As we will see, the mechanism applied to obtain this representation from the output of linguistic analysis has underwent a complete redesign to increase the coverage of the TA. A consequence of this redesign is that in its final version, the JSON analyses produced by the TA include a *refs* attribute in all incidents that contains disambiguated references to knowledge resources -BabelNet, OSM and GeoNames. While the basic version included DBPedia references as a mean to link to additional information about the entities detetected during analysis, references to BabelNet now play a pivotal role in determining the ontological type of each entity. In the example of Figure 10, references, that have been added to BabelNet, have been assigned to each entity, and references to OSM and GeoNames have been assigned to the entity of type *Location*.


```
"fire_0": {
  "id": "fire_0",
  "label": "fire",
  "type": ["Fire"],
  "refs": ["bn:00034625n",“bnLabel:WIKT:EN:flame"],
  "participants": [
    {"role": "state","participant": "danger_0”},
    {"role": "participant","participant": "danger_0"},
    {"role": "location", "participant": "el_Saler_0"}
  ]
},
"forest_0:": {
  "id": "forest_0",
  "label": "forest",
  "type": ["Forest"],
  "refs": ["bn:00035868n",”bnLabel:WN:EN:forest"],
},
"danger_0": {
  "id": "danger_0",
  "label": "danger",
  "type": ["Risk"],
  "refs": ["bn:00030747n",”bnLabel:WN:EN:jeopardy"],
},
"el_Saler_0": {
  "id": "el_Saler_0",
  "label": "el Saler",
  "type": ["Thing"],
  "refs": [
    "bn:04600242n","bnLabel:WIKIDATA:ES:Saler",
    "geonamesId:2518120",
    "geonamesLabel:El Saler","osmLabel:el Saler"
  ],
  "location": {
    "latitude": 39.382862,
    "longitude": -0.329648,
    "label": "el Saler"
  },
}
```


Figure 10: JSON analysis resulting from sentence "Danger of forest fire in El Saler"

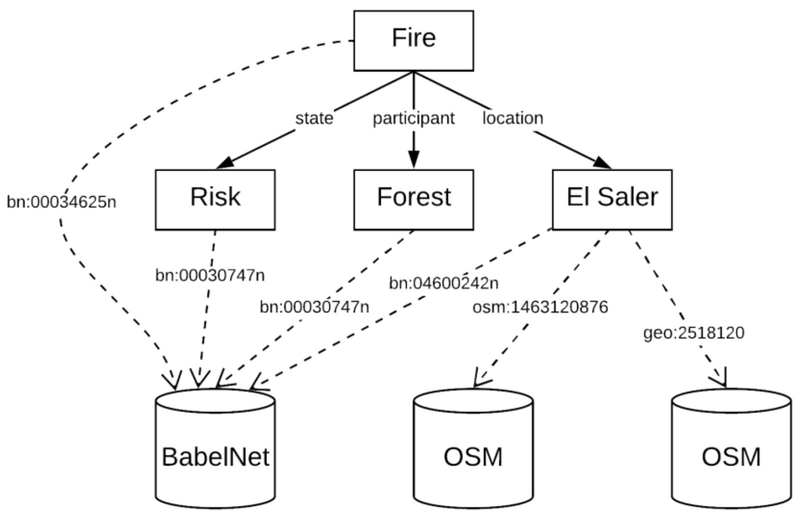



Figure 11: Diagram of the analysis resulting from sentence "Danger of forest fire in El Saler"

### 5.2.2 Knowledge extraction pipeline

The rudimentary method for knowledge extraction in the basic version of TA has been replaced by a fully-functional multilingual IE pipeline that addresses several tasks to abstract away from linguistic and language-specific representations. The pipeline, as shown in Figure 12, starts by applying Named Entities Recognition (NER) and terminological concept detection separately on the input text. For NER we use spaCy[11], while for concept detection we use a tool developed by UPF within the scope of the TENSOR and beAWARE projects. Both tools apply models trained on large corpora to detect potential mentions of named entities and domain-relevant concepts.

Named Entities (NE) annotations of type location are used as input to our geolocation tool, developed within beAWARE, which looks up potential locations in geographical databases using the text annotated as a location and chooses the best location. NE annotations of any type, concept annotations and single words are used as input to the disambiguation tool, developed for the TENSOR and beAWARE projects. This tool looks up the annotated text in BabelNet, a multilingual lexicographical database, and determines the best meaning for the text. NER and concept detection tools serve as means to detect potential locations and multiwords, and thus avoid computationally expensive brute-force approaches when looking up locations and meanings in the subsequent geolocation and disambiguation steps. In other words, instead of looking up in the external resources -BabelNet, OSM, Geonames- all combinations of consecutive words in the input texts, we restrict the look up to promising words or multiwords according to the NER and concept detection models.

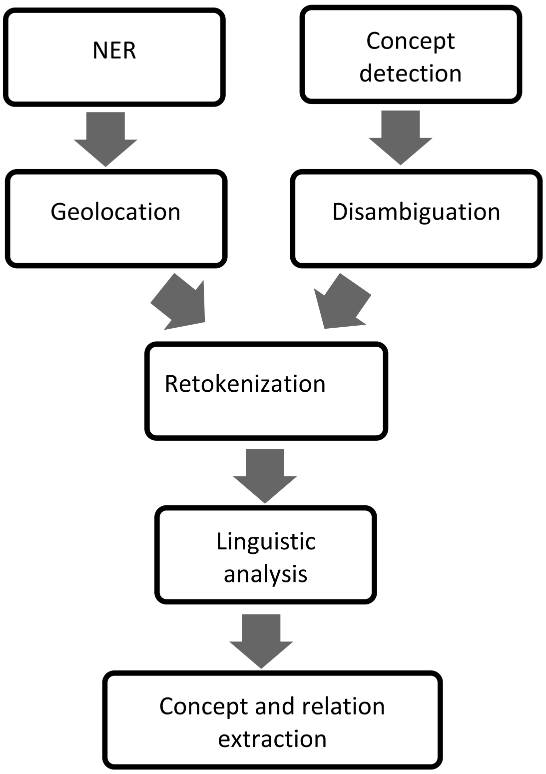



Figure 12: Knowledge acquisition pipeline, including the linguistic analysis step

[11] https://spacy.io/

The annotations produced by the geolocation and disambiguation tools may overlap. For this reason, and as a preparatory step towards obtaining a unified knowledge representation, these annotations are reconciled and marked as single tokens by a rule-based retokenization component. This component performs the following steps:

1. Mark as a single token all geoloated multiwords
2. Discard any annotations produced by disambiguation that overlap with geolocated multiwords
3. Mark as a single token all multiwords assigned with a single meaning by the disambiguation component

The first step may result in expresions like 'El Saler' being merged into a sinlge token if a location for this name was found in either GeoNames or OSM. The second will discard any BabelNet annotations that span over part of the words in 'El Saler', thus keeping a meaning for 'El Saler' if found in BabelNet but discarding meanings for just the word 'Saler'. The last step marks as a single token any group of consecutive words assigned with the same meaning, e.g. 'emergency vehicle'.

The retokenized text is then passed as input to the linguistic anlysis step, which will produce a deep syntactic graph where deep dependency relations hold between pairs of words or multiwords in the original text. The rule-based concept and relation extraction module, also developed specifically for beAWARE, simplifies the deep syntactic graph of each sentence into one or more n-ary relations expressed in conceptual -ontological- terms.

Our components for concept detection, geolocation and disambiguation are described in greater detail in their respective subsections below.

**Concept detection**

In the following we describe the updated method for terminological concept detection. The main difference between the basic method presented in D3.3 and the advanced one is that we have applied supervised learning methods to train a statiscal model. In the following we describe how we trained this model.

Firstly, we use the previous basic version of the concept extraction tool for relatively weak annotation of a large corpus of domain-independent texts, and then adapt a summarization-oriented pointer-generator model proposed in (See, Liu, and Manning 2017) to be trained on the annotated data to spot the concepts in a sentence. As it is inherent in distant supervision techniques, this training allows the model to overcome errors provided by the not perfect annotator. In what follows, we describe the details of the overall method.

The algorithm of compilation of the weakly annotated training corpus consists of the following steps:

1) determining part-of-speech tags for a given text and selecting potential parts of noun phrases comprising two open class lexical items (terms);
2) assessing the distinctiveness of each selected part depending on its position within a list of similar term-co-occurrences (differ by one term) ordered by frequencies;
3) combining intersected highly distinctive parts into concepts and leaving the remainder as separate concepts if they form noun phrases by themselves;
4) applying statistical NER model to detect additional out-of-vocabulary multi-word expressions;

5) eliminating parts of detected concepts that have an overlap with named entities;
6) compiling an output list of non-overlapped and non-nested concepts including named entities as the result to be used as a target sequence in seq2seq learning.

For the first step, possible part-of-speech patterns for matching complex noun phrases as candidates for parts of concepts have been designed. They include, but are not limited to, the patterns introduced in (Cordeiro et al., 2016). The whole set of patterns $\bigcup c_i$ is the following: N-N, J-N, V-N, N-J, J-J, V-J, N-of-N, N-of-DT-N, N-of-J, N-of-DT-J, N-of-V, N-of-DT-V, CD-N, CD-A, where V is limited only to verbs of types VBD|VBG|VBN. Each pattern is used for extracting n-grams with two open class lexical items and at most two auxiliary tokens in between them that are parts of potential concepts. The concepts are formed using combinations of these parts.

The distinctiveness of selected n-grams is assessed on word co-occurrences from the Google Books dataset[12]. Given an n-gram $T_1A_1A_2T_2 \in c_k$, where $T_1$ and $T_2$ are terms ("meaningful" words out of a stop-word list) and $A_1$ and $A_2$ are optional auxiliary tokens, and $c_k$ is a particular kind of pattern, we use it as a point on a continuous function passing through frequencies of a set of similar n-grams $\bigcup_{c_k} T_1A_1A_2T_j$, taken in a descending order of their frequencies, to find the gradient of decrease of the function. In other words, we check how strong the prominence of n-grams differs from the prominence of their neighbors. In case an n-gram is located within a long range of equally prominent n-grams we do not consider it as a potential part of a concept as it does not possess the notable distinctiveness inherent in concepts especially in ones with a direct meaning. The thresholds $Q_{min1}$ and $Q_{min2}$ for a minimum allowed angles of a slope among the sets $\bigcup_{c_k} T_1A_1A_2T_j$ and $\bigcup_{c_k} T_hA_1A_2T_2$ are defined using the development set. The $Q_{min1}$ concerns the maximum value of two angles and $Q_{min2}$ – the minimum value of them.

Once the potential parts of concepts are detected, we join those that share common tokens and iteratively drop the last token in each grouped sequence of tokens if it is not a noun, in order to end up with complete noun phrase concept candidates. Afterwards, we take all nouns and numbers in a text as single-word candidates and drop those that have already been included in the compound candidate concepts.

The described frequency-based criterion for selecting the parts of concepts allows the detection of prominent commonly used compound terms and named entities. Some out-of-vocabulary (OOV) concepts such as the capitalized proper nouns might be caught by a statistical named entity recognizer (Lample et al., 2016, Honnibal and Montani, 2017). For this reason, and as mentioned before, a state-of-the-art NER tool is applied with a successive elimination of parts of previously found candidate concepts included in extracted named entities to leave non-nested and non-overlapped n-grams for the outcome.

The above method for labeling large datasets within distant supervision might be substituted by a dictionary-based approach, e.g., DBpedia Spotlight (Daiber et al., 2013). However, dictionary-based approaches are worse in detecting non-named multi-word concepts. This negatively influences the outcome of training but improves results if used as a compliment.

[12] https://books.google.com/ngrams/

Therefore, we use a combination of outcomes of two models trained on differently annotated datasets as a final result.

We chose a bidirectional LSTM (Luong et al., 2015) with a copy mechanism (Gu et al., 2016; See et al., 2017) as a model for generating the sequence of concepts. In particular, we use the biLSTM realization of the OpenNMT toolkit (Klein et al., 2018) that enables a pointer which allows copying tokens from the reference text.

The input parts of training examples are subsequent pairs of tokens and their POS tags, separated by a white space (e.g., 'conceptA NN is VBZ followed VBD by IN the DT second JJ concept NN'). The target sequence is a list of concepts separated by a special token (e.g., 'conceptA * second concept'). The sequences are taken from sentences with a sliding overlapping window of a fixed length, which are prolonged in case of incomplete candidate concepts at the end.

The trained model is applied to unseen sentences, which are also split into sequences of tokens with an overlapping window of the same size. Finally, determining the positions in a raw text is performed since the output format does not imply including offsets. In particular, we find all possible matches for all detected concepts and then iteratively select non-nested concepts from the beginning to the end of the sentence, giving priority to the longest in case more than one concept starts with the same token.

We used a snapshot of Wikipedia provided by Schenkel et al. (2007) for training several models. The provided semantic annotation was not taken into account. Rather, we used only raw texts of pages and texts of the pointers to other pages as ground truth concepts.

Several subsets were selected from the collection of Wikipedia pages: 220K pages as a dataset to be annotated and used for training, 30K pages for internal deep learning validation steps, 7K pages as a validation set for choosing parameters of distant supervision and selecting the best model among several trained with different parameters, and 7K pages as a test set.

The grid search was applied in order to find the best combination of parameters $Q_{min1}$ and $Q_{min2}$ from the three possible angles of a slope corresponding to the different levels of the distinctiveness of a concept described above: $85°, 60°, 0°$. SpaCy NER (Honnibal and Montani, 2017) was used to expand the set of detected candidate concepts with named entities. A separate annotation with DBpedia Spotlight was conducted in order to complement and improve the outcomes of differently trained models.

The training was performed several times with the different amount of resources: networks of two layers with 10K and 20K steps of training and of three layers with 10K and 100K steps for which their different checkpoints were tested. The models that reached the highest $F_1$-score on the validation set were chosen. They are bidirectional LSTMs with 2 layers and 18K/20K training steps on data annotated by our algorithm and with 3 layers and 80K/100K training steps on the annotation obtained with DBpedia Spotlight ($F_1$-score is 0.72 and 0.71 correspondingly). Tests on generic data showed that two models trained on differently annotated data applied together improved the F1-score by about 10% in comparison to the values obtained with the individual models.

**Geolocation**

For location extraction, we created direct and inverted search indexes over locations from two geographical databases Open Street Maps[13] and GeoNames[14]. In order to have a memory efficient search, first, we processed the geo-data and stored them as disk space level structures to operate with the queries without loading data into RAM. Second, we pruned these structures and organized them into a set of smaller pilot-oriented indices. This procedure also helped to disambiguate entities with the same names located in and out of the region of interest.

Identification of locations is carried out by processing the input text to create a search request and subsequent querying the domain-dependent search index to obtain coordinates of mentioned objects. Textual analysis of social media texts is based on named entity recognizer and linguistic dependency-based patterns to identify candidate location mentions, in synergy with the DBpediaSpotlight links, in order to determine whether a mention refers actually to a location or not. In particular, SpaCy[15] was selected for extracting named entities as it provides a robust state-of-the-art solution for this task. The choice of SpaCy as a promising NER tool is supported by the evaluation presented below in the corresponding section.

The overall algorithm of detecting location candidates to form search query consists of the following steps: i) if a place-indicating mention, such as “park”, “avenue”, “highway”, etc. is linked via a NAME dependency to a proper name, then their concatenation is marked as a location; ii) if a DBpedia resource link has been obtained for a, single- or multi-word, mention, and among its DBpedia types, the classes dbo:Place or dbo:SpatialThing are included, then the mention is marked as a location; iii) likewise, if the mention under consideration has been tagged by the NER tool as a location.

Special mention normalization that concerns transforming hash-tags to a proper form and also facilitates case-insensitive search is applied. It is necessary due to the nature of short spontaneous messages with generally erroneous spelling.

Apart from coordinates of queried objects, the structure of search index allows returning various lexicalizations of entities needed for multilingual report generation.

**Disambiguation**

The basic version of the text analysis pipeline used DBPedia Spotlight[16] to determine the semantics of names and terms. As explained this tool does not support Greek and is limited to meanings in DBPedia, thus excluding word senses for many common words, in particular those of non-nominal words. For this reason, we decided to replace it with our own disambiguation algorithm applied to BabelNet 4.0.1. A version of this component tailored for Italian and English texts belonging to the 2nd pilot was described and evaluated in D7.6. Since then, the disambiguation component has been extended to also support Spanish and Greek, and adapted to 3rd pilot texts. The latter involved adding a manually compiled list of synsets - listed in Table 4**Error! Reference source not found.**- that express important meanings for the

[13] https://www.openstreetmap.org/

[14] https://www.geonames.org/

[15] https://spacy.io/

[16] https://www.dbpedia-spotlight.org/

3rd pilot, while the former involved incorporating a few language-specific resources to the component.

Fasttext word embeddings[17] are used for comparing candidate meanings based on the glosses found in BabelNet, and to compare them to the text being analyzed. The algorithm also uses lists of stop words for each language taken from a public repository[18]. Finally, we created memory-efficent versions of BabelNet implemented by collecting only the information required for disambiguation and generation in WP5, and storing this information using byte buffers and the GNU Trove library[19]. This compact dictionary, one for each of the beAWARE languages, are used to greatly reduce execution time and memory print of the TA and RG services.

Table 4: Context meanings used for 3rd pilot

| BabelNet sysnet | Ontology concept |
|---|---|
| bn:03411833n | natural disaster |
| bn:00030525n | emergency |
| bn:00034623n<br>bn:00035870n | Fire |
| bn:00084270v | Burn |
| bn:00036806n | Smoke |
| bn:00043416n | Heat |
| bn:00002216n | Wind |
| bn:00045190n | Humidity |
| bn:00028483n | Precipitation |
| bn:00079484n | Valencia |
| bn:00030747n | Risk |

**Concept and Relation extraction**

The starting point for concept and relation extraction is a UD-based deep syntactic graph where nodes correspond to either single or multiple words from the original text that may be associated with disambiguated meanings and/or locations. The end goal is to produce n-ary relations expressed in terms of classes and relations in the beAWARE ontology. Given that the ontology used in beAWARE has been newly developed for the domain of emergency situations, it wasn’t possible to reuse or adapt exisiting tools for the step from UD graphs to the ontology.

[17] https://fasttext.cc/

[18] https://github.com/stopwords-iso

[19] http://trove4j.sourceforge.net/html/overview.html

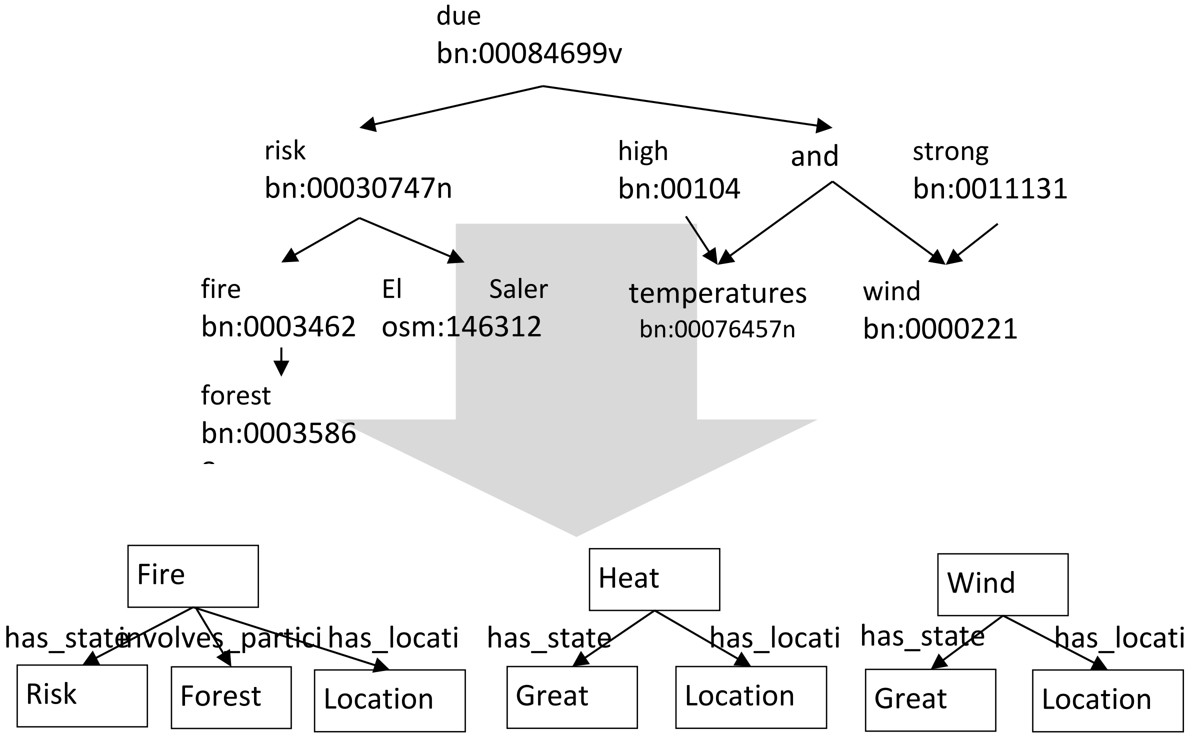



Figure 13: Relation extraction from UD graph for sentence "Risk of forest fire in the El Saler due to high temperatiutures and strong wind"

Figure 13 shows the UD graph for a sentence in its top half, and three relations that can be extracted from it its bottom half. In order to extract these relations, we implemented a two-step extraction procedure. In a first step, language-independent meanings, either BabelNet synsets or locations, are mapped to ontological classes. This mapping was obtained semi-automatically by manually determining a canonical BabelNet synset for each ontology class and then extracting synsets connected to them in BabelNet knowledge graph via hyponymy or derivationally related relations. The resulting sets of synsets, one per class in the ontology, are then used to deterministically map nodes in the UD graph to instances of ontology classes based on the disambiguated BabelNet synsets associated with the nodes.

In a second step, a relation is extracted from each instance of an Incident class or subclass. First, any locations found in the UD graph are added as locations of each incident. In the example shown in Figure 13, the nodes for words 'fire', 'temperatures' and 'wind' result in three relations concerning instances of classes *Fire*, *Heat* and *Wind* respectively. For each incident instance, all dependents of the corresponding node in the input graph that have been mapped to subclasses of *Vulnerable Object* are added as participants in the relation, i.e. as objects impacted by the event indicated by the incident. In our example, this results with an instance of class *Forest* being added as a participant of the fire relation. Finally, governors of each incident node that are mapped to subclasses of *State* are also added to the respective relations. The relation concerning a fire is added a state of type *Risk*, and the relations concerning heat and wind are added instances of *Greater* as state.

Unlike the rest of ontological classes, the detection of instances of *Risk* is not triggered by lexical cues but by morhosyntactical features of the UD graph. More precisely, *Risk* is instantiated for a given incident if there is a verb governing the word that indicates the incident and this verb has either conditional mood -in English, e.g. “there may be a fire”- or subjunctive tense -in Spanish and Italian, e.g. “Si se declarase un incendio”.

# 6 Crisis Classification and Risk Assessment

## 6.1 Overview

In the framework of beAWARE project, the Crisis Classification component is responsible for providing accurate notifications to crisis managers, authorities and first responders for a potential upcoming extreme weather climate event. Furthermore, it should identify and classify the severity of the ongoing crisis by monitoring it and providing aggregated real-time information to decision-makers. In order to accomplish these goals, two modules have been developed so far, namely, the *Early Warning* module and the *Real-time Monitoring and Risk Assessment* as already described in Deliverable D3.1.

Particularly, the *Early Warning* module consolidates technologies for a timely forecasting analysis aiming to estimate and classify the severity level of a forthcoming crisis weather event by covering the flood, fire and heatwave use cases. Towards on this direction, it obtains forecasts from weather prediction models (e.g. HIRLAM etc.), for hydrological prediction models for river water level (e.g. AMICO) and estimation of fire danger (e.g. Canadian Forest Fire Weather Index from EFFIS). The aggregated data are forwarded to the beAWARE dashboard and to the Public Safety Answering Points (PSAP) in order to inform decision-makers for the upcoming crisis conditions, so as to enhance their awareness and ensure the civil preparedness.

Furthermore, the *Real-time Monitoring and Risk Assessment* module enables the crisis managers to identify and classify the emergency event in terms of its severity, by obtaining real-time or 'near' real-time data, such as weather observations, river water level measurements, etc. Moreover, this module is equipped with enhanced functionalities so as to fuse heterogeneous data and estimate dynamically the severity level by the acquired information from textual and multimedia analysis (e.g. images, videos, UAVs etc.).

In the following subsections, the final version of these modules will be presented so as to capture the user requirements and specifications for the three beAWARE's use cases, namely for flood, forest fire and heatwave. It is worth to note that some of the functionalities have already been mentioned in the previous deliverables, mostly in D3.1, and are considered as part of the 1st and 2nd prototype of the Crisis Classification component.

## 6.2 The Early Warning module

In the pre-crisis phase, the Early Warning module acquires the forecasting data for the Region of Interest (RoI) in regular and specific time intervals. The main objective is to assess the upcoming crisis's severity level in the whole RoI and/or in specific areas, to give timely appropriate warnings to the authorities and crisis managers. The whole methodological framework for the three use cases has been extensively illustrated in the Deliverable D3.1 and deployed in the 1st prototype. The proposed approach had been enriched in the 2nd prototype of the Crisis Classification component in which the utilisation of the flood hazard/risk maps had been integrated into the beAWARE platform (see also in Deliverable D7.6). Briefly, the *Early Warning* module includes the following steps:

**Step 1. Data Acquisition.** It includes the processes to grab stream of time series forecasts from various prediction models and sources:

a. For the flood use case, the forecasts are generated as the outcome of the AMICO hydrological and hydraulic prediction model. Each Flood forecast includes time series of forecasted water level for a set of 304 river sections. For each of these, the forecasted water levels are given in a frequency of one hour whereas the time series covers up to 54 hours (HIRLAM's maximum horizon of forecast) from the data of emission.
b. For the fire use case, the HIRLAM prediction model provides the weather forecasts of air temperature, humidity, precipitation, wind speed and direction at the region of interest (Valencia). The predictions are given by an hourly time step and the forecasting period covers up to 48 hours ahead. The assessments of fire danger rely on the Canadian Forest Fire Weather Index provided by the EFFIS for a period of 9 days.
c. For the heatwave use case, the HIRLAM prediction model provides the weather forecasts of air temperature, humidity and wind speed/direction at the region of interest (Thessaloniki) as above.

**Step 2. Data Fusion**. It includes the processes to estimate:

a. the estimated level of the crisis based on water level forecasts for the flood use case and on the Canadian Forest Fire Weather Index and fire danger of the fire use case. Additionally, the Discomfort Index is estimated by the utilisation of the weather forecasts in the heatwave use case,
b. the risk of flood in specific areas (polygons) near the river section based on the data extracted from the provided risk maps. The risk maps are furnished by AAWA in the Early Warning module,
c. the crisis level in each group of river reach and the overall flood crisis level in the flood use case. Similarly, in the other two use cases, the severity assessments of the upcoming fire/heatwave extreme events are provided per forecasting day.

**Step 3. Data Analysis Presentation.** It includes the processes of the presentation of the data analysis outcomes aiming to support the decision-making process. The results are illustrated in two ways:

a. the appropriate messages are created and forwarded to the PSAP in order to present the results onto the map,
b. the appropriate messages are created and forwarded to the beAWARE dashboard in order to present the results via interactive charts (lineplots, barplots, gauge and traffic light charts).

The following figure (Figure 14) illustrates an overview of the Early Warning module and its interaction with other beAWARE component or external resources. The Early Warning module connects with the FROST-server and beAWARE Knowledge Base to obtain the sensing data and with the beAWARE PSAP/map and dashboard to forward the results of the analysis. Furthermore, to obtain the weather predictions from the Early Warning module a direct connection with FMI's Open Data API is established.

Finally, the estimations of the Forest Fire Weather Index along with the fire danger assessments are retrieved from the EFFIS API.

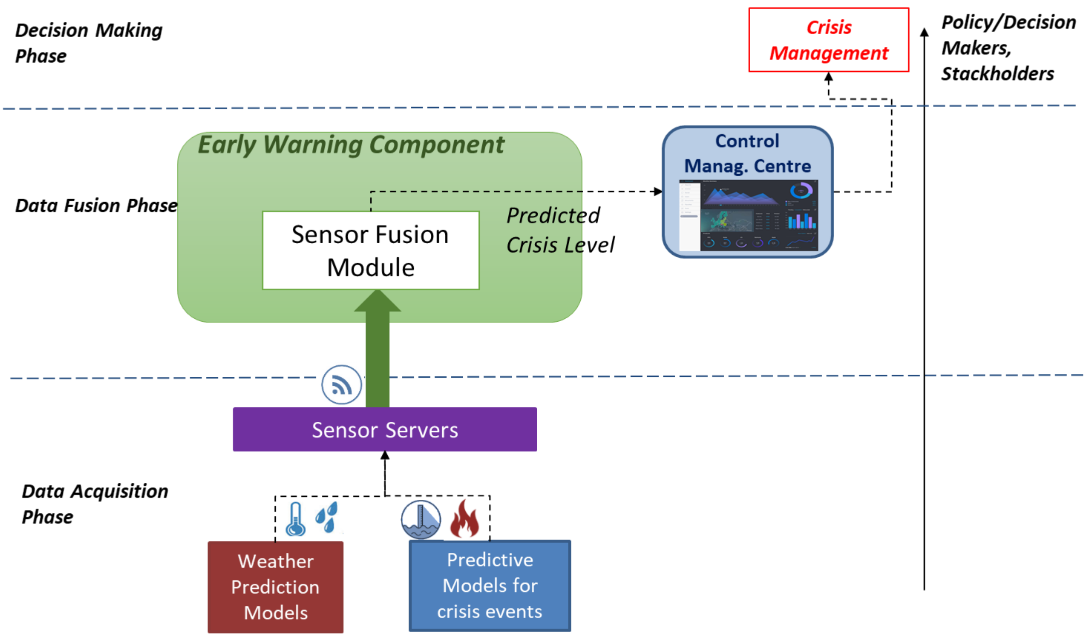



Figure 14: Overview of the Early Warning module

## 6.3 The Real-time Monitoring and Risk Assessment module

During the emergency phase, the *Real-Time Monitoring and Risk Assessment* module is triggered aiming to track and inform authorities and crisis managers regarding the evolution of the natural crisis event (i.e. flood, fire, heatwave). It classifies the ongoing disruptive event in terms of its severity level by taking into consideration heterogeneous real-time data from the region of interest. In Figure 15 an overview of the *Real-Time Monitoring and Risk Assessment* module is presented. The following steps are carried out:

**Step 1. Data Acquisition**: includes the processes to obtain real-time observations at sensors which are located in specific weather stations. The measurements vary depending on the use case. In flood pilot, the scope is very concise focusing on the status of the current water level and the amount of precipitation. On the other hand, in the other two use cases meteorological real-time observations, such as air temperature, humidity, precipitation and wind speed/direction, are most useful, as their exploitation provides accurate estimations of the severity level of the evolving fire or heatwave event.

**Step 2. Data/Information Fusion phase**: both a multi-layer fusion approach that encompasses the data/information fusion layer and a decision fusion layer have been developed in order to combine the heterogeneous data and assess the risk/severity level of the ongoing crisis. In the *Sensor Fusion* module, weather observations and other useful information such as the river water level which are obtained from the sensors located in the region of interest, are combined, aiming to assess the observed crisis severity level. This process has been described in more detains in the Deliverable

D3.1 and it has already been implemented in the 1st prototype of Crisis Classification component. Specifically, a rule-based fusion approach which relies on linear weighted fusion has been adopted. In the second layer, the *Decision Fusion* module obtains information dynamically from the field of interest and attempts to estimate the ongoing crisis risk/severity level. Specifically, it receives input from a variety of beAWARE modalities (sources), such as from citizens and first responders, who report data via incident reports through their mobile application, which are preprocessed through multimedia analysis (the image, video and audio analysis components) as well as the textual analysis component. The goal is to dynamically estimate the level of risk/severity of the evolving crisis event and to inform the crisis managers and decision makers regarding the current conditions in the region of interest.

**Step 3. Data Analysis Presentation.** As in the Early Warning module, this step encompasses the processes of the presentation of the data analysis outcomes aiming to support the decision-making process. Specifically, the risk/severity level of the groups of incidents and in the whole region of interest are updated accordingly based on the imported information and they are presented in the PSAP/map as well as in the suitably created visuals in the beAWARE dashboard.

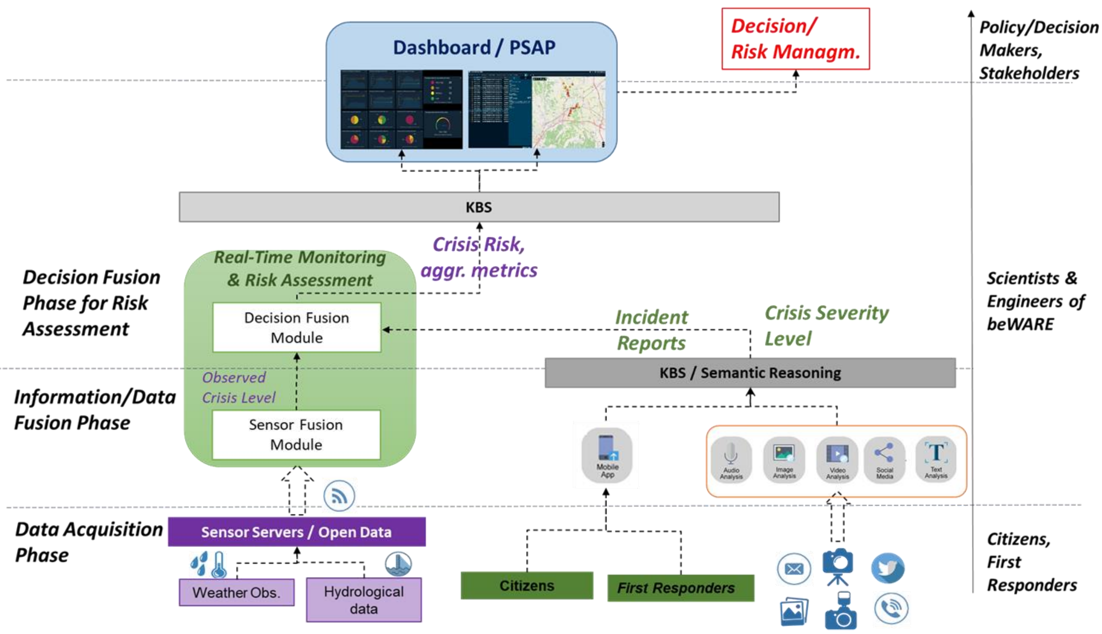



Figure 15: Overview of the *Real-Time Monitoring and Risk Assessment* module

Two modifications of the above general approach have been implemented in the framework of 2nd and 3rd prototype of the Crisis Classification module so as to meet the special needs of the flood and fire pilot. In the following subsections these alternatives are described in more details.

### 6.3.1 The Real-Time Monitoring and Risk Assessment module in the Flood pilot

The goal of this approach is to dynamically assess the risk of the ongoing flood crisis event by taking under consideration real-time input, which is obtained directly from the first responders and/ or citizens in the region of interest. The following algorithm employs local information from a citizens' perspective in order to estimate the current crisis risk based on the details provided by them regarding the people in danger, the contiguous buildings or other historical assets etc. The flood risk assessment algorithm is created in co-operation with AAWA, a beAWARE partner. Briefly the steps of the algorithm are the following:

**Step 1.** **Data Acquisition**: includes the processes to request the data related with the evolution of the flood crisis from citizens' mobile application via appropriately designed reports. These are retrieved from the beAWARE Knowledge Base Ontology.

**Step 2.** **Decision Fusion:** estimates the risk assessment by exploiting the received information from the citizens. It includes the calculations of hazard, exposure, vulnerability and finally the hydraulic risk and severity of each incident. If it is needed, the obtained information is enriched with data that is extracted from GIS files represented in historical hazard and risk maps. Those files have been stored in beAWARE's geoServer and are related with historical river water level observations, the exposure of assets and their vulnerability in the Vicenza region as well as the severity level and risk estimations.

**Step 3.** **Store** the incident report and results of the analysis to the local database and **create** the appropriate messages, which are sent to the PSAP in order to update the status of each incident.

**Step 4.** **Calculate** dynamically the accumulated **Crisis Risk** and **Severity Level** relying on the risk/severity of all obtained incident reports.

**Step 5.** **Store** the results to the local database and **create** the appropriate messages for the Dashboard in order to update the corresponding plots.

The important issue of the above algorithm is the data that the system receives from the citizens that are in the field of interest. They play a significant role by operating as actors and "real-time" sensors from the field. Initially, they should use the specifically designed mobile application in order to report valuable input to the Crisis Classification component. Particularly, they can choose the *Category* field (generic flood report or void) and the *Hazard* field, which provide details related with the estimated water level in the flooded area (Table 5).

Table 5: Hazard field options. W.L. indicates the estimative water level in flooded area

| # | Hazard field option (optional)[20] |
|---|---|
| 1 | W.L. ≤ 0.25m |
| 2 | 0.25m < W.L. ≤ 0.5m |
| 3 | 0.5m < W.L. ≤ 0.75m |
| 4 | 0.75m < W.L. ≤ 1m |
| 5 | W.L. > 1m |

[20] Note: only one selection allowed

Furthermore, citizens should select options from the *Exposure* field which indicates which are the assets that are in danger in the flooded area. They could select the option "People involved", indicating the presence of people in the field **and/or only one** of the other options as presented in the following table:

Table 6: Exposure field options

| # | Exposure field option (optional) |
|---|---|
| 1 | People involved |
| 2 | Buildings (private, public, economic, industrial and historical buildings, touristic) |
| 3 | Infrastructures (roads, etc.) |
| 4 | Rural area |
| 5 | Camping ground, sport field |
| 6 | Natural or semi-natural area, unproductive areas |

When the beAWARE system receives an incident report the *Real-Time Monitoring and Risk Assessment* module is triggered. Specifically, the *Decision Fusion* module is activated in order to assess the Risk of a particular incident using the following formula:

$$R = \frac{10 * H * E_p * V_p + H * E_e * V_e + H * E_a * V_a}{12}$$

where:

- $H$ indicates the **Hazard**
- $E_p$, $E_e$, $E_a$ indicate the exposure of people ($E_p$), exposure of economic activities ($E_e$), exposure of environment and cultural assets ($E_a$). All these indicators take values between 0 and 1.
- $V_p$, $V_e$, $V_a$ indicate the vulnerability of people ($E_p$), vulnerability of economic activities ($E_e$), vulnerability of environment and cultural assets ($E_a$). All these indicators take values between 0 and 1.
- $R$ indicates the risk of the incident, $0 \leq R \leq 1$.

Based on the risk value, the level of severity/risk is estimated as follows (Table 7):

Table 7: Classify Risk (R) into Severity Levels

| Risk R | Level of Risk/Severity |
|---|---|
| $0 \leq R < 0.2$ | Minor |
| $0.2 \leq R < 0.5$ | Moderate |

| Risk R | Level of Risk/Severity |
| --- | --- |
| $0.5 \leq R < 0.9$ | Severe |
| $0.9 \leq R \leq 1.0$ | Extreme |

In the following paragraphs, a description of the manner that the above quantites can be estimated is presented.

**Calculating the Hazard**

If, in the mobile application, the citizen choose an option of the hazard field that indicates his/her estimation for the current river water level, then the Hazard (H) value as well as its classification is calculated as follows (Table 8):

Table 8: Hazard values and categories related with the estimated water level

| Estimate WL | Hazard H Value ($0 \leq H \leq 1$) | Description |
| --- | --- | --- |
| W.L. < 0.5m | 0.4 | Low hazard |
| 0.5 ≤ W.L. < 1m | 0.8 | Medium hazard |
| W.L. ≥ 1m | 1 | High hazard |

If the hazard field is void, then the estimation of the Hazard should rely on the pre-defined Hazard maps which have already been created by historical data and are presented in GIS shapefiles. Based on the coordinates of the incident, the module calculates the appropriate area (polygon) in the shapefile and extracts the height of the water level in meters. Then, according to the above table (Table 8) the Hazard value for the particular incident report is obtained.

**Calculating the Exposure**

Three different categories of exposure are considered in the proposed risk assessment algorithm. Hence it estimates the exposure of people ($E_p$), the exposure of economic activities ($E_e$) and the exposure of environment and cultural assets ($E_a$). The values of these quantities fluctuate between 0 and 1.

If the exposure field in the mobile application have been filled out, the values of $E_p$, $E_e$ and $E_a$ can be estimated, as shown in the following table (Table 9):

Table 9: Exposure values per category of exposure ($E_p$, $E_a$, $E_e$) and exposure field

| Exposure field | $E_p$ | $E_e$ | $E_a$ |
| --- | --- | --- | --- |
| Buildings | 1 | 1 | 0.9 |
| Infrastructures | 0.5 | 1 | 0.2 |
| Rural area | 0.3 | 0.6 | 0.7 |
| Camping grounds | 1 | 1 | 0.1 |
| Natural or semi-natural area | 0.1 | 0.1 | 1 |

Otherwise, if a citizen has not filled in the exposure field, then the estimation of the exposure ($E_p$, $E_e$, $E_a$) can be taken from AAWA's GIS shapefiles. The goal is to identify the closest asset to the coordinates of the reported incident.

**Calculating the Vulnerability**

Similarly, as in the case of the Exposure, in the proposed risk assessment algorithm the vulnerability consists of three different categories, namely the vulnerability of people ($V_p$), the vulnerability of economic activities ($V_e$) and the vulnerability of environment and cultural assets ($V_a$). The values of these quantities fluctuate between 0 and 1.

The vulnerability depends on the values of both hazard and exposure field. If the citizen has filled these fields via mobile application then the estimation of the three vulnerability categories is quite straightforward:

a. Vp – vulnerability of people

It is necessary to calculate a parameter named *Debris Factor* (*DF*), related to the presence of floating materials, using the values of the following table:

Table 10: Debris Factor (DF) depends on the estimated water level and element at risk

| | Debris Factor | | |
|---|---|---|---|
| Estimate Water Level | If element at risk is Rural Area | If element at risk is Natural or semi-natural area | Urban area (if element at risk is different from rural or natural area) |
| W.L. < 0.25m | 0 | 0 | 0 |
| 0.25 ≤ W.L. < 0.75m | 0 | 0.5 | 1 |
| W.L. ≥ 0.75m | 0.5 | 1 | 1 |

Then, the *Flood Hazard parameter* (FHR) is estimated based on the following formula:

$$FHR = 1.5 * W.L. + DF$$

Then the Vp values are defined as below:

Table 11: Vulnerability of people depends on the *Flood Hazard values*

| Flood Hazard parameter (FHR) | $V_p$ ($0 \leq V_p \leq 1$) |
|---|---|
| FHR < 0.75 | 0.25 |
| 0.75 ≤ FHR < 1.25 | 0.75 |
| FHR ≥ 1.25 | 1 |

b. Ve – vulnerability of economic activities

The vulnerability of economic activities relies on the element at risk and the estimated water level. Thus, its value in each case is defined following the table below:

Table 12: Vulnerability of economic activities depends on element at risk and estimated water level

| If Element at risk = Buildings | |
|---|---|
| W.L. (m) | Ve |
| W.L. < 0.5 | 0.25 |
| 0.5 ≤ W.L. < 1 | 0.75 |
| WL ≥ 1 | 1 |
| If element at risk = Caming grounds | |
| W.L. (m) | Ve |
| W.L. < 0.25 | 0.25 |
| 0.25 ≤ W.L. < 0.75 | 0.75 |
| W.L. ≥ 0.75 | 1 |
| If Element at risk = Infrastructures | |
| W.L. (m) | Ve |
| W.L. < 0.25 | 0.25 |
| 0.25 ≤ W.L.< 0.5 | 0.75 |
| WL ≥ 0.5 | 1 |
| IF Element at risk = Rural area | |
| W.L. (m) | Ve |
| W.L. < 0.5 | 0.5 |
| W.L. ≥ 0.5 | 1 |
| IF Element at risk = Natural or semi-natural areas | |
| W.L. (m) | Ve |
| W.L.<0.5 | 0.25 |
| W.L. ≥ 0.5 | 0.5 |

c. Va– environments and cultural assets
In this version of the risk assessment algorithm, we choose to maximise the vulnerability of cultural assets giving the value 1 in this factor.

Table 13: Vulnerability of environments and cultural assets

| Va |
|---|
| 1 |

In the case that the citizen has not filled the hazard and/or exposure field, then the algorithm utilizes the appropriate GIS shapefiles so as to extract the acquired values for the vulnerability components.

**Estimate the accumulated Crisis Risk and Severity Level**

In Step 4 of the above algorithm, the assessment of the accumulated Crisis Risk and the overall Severity Level is carried out dynamically, relying on the Risk of the incidents which took place in the field. In the 2nd prototype, two rule-based decision fusion approaches have been adopted and evaluated, specifically the majority voting and the linear weighted fusion (Atrey, M. Anwar, El Saddik, & Kankanhalli, 2010).

Let $R_i$, $1 \leq i \leq N$, presents the risk of the i-th incident report and N indicates the number of incident reports which are received so far from the component. In majority voting fusion, the final decision (risk) is the one where the majority of the incidents have a similar severity level.

In the linear weighted fusion approach, the final risk (R) is estimated based on the following formula:

$$R = \sum_{i=1}^{N} w_i * R_i$$

In our case, we modify slightly the above formula in order to give bigger significance to the higher severity levels. Thus, we adopt the notion of generalised (power) mean with p=4. The levels of risk/severity (Table 7) correspond to a scale from 1 (= Minor) to 4 (= Extreme). So, as $N_k$ indicates the number of incidents' risk that belong to the k-th severity level. Then, the weight $w_k$ can be defined as

$$w_k = \frac{N_k}{N}, \qquad k = 1, \dots, 4$$

and the accumulated Risk over all incidents can be determined by the following formula:

$$R_{acc} = \left\lceil \sqrt[p]{\sum_{k=1}^{4} w_k * scale_k^p} \right\rceil = \left\lceil \sqrt[p]{\frac{N_4\, 4^p + N_3\, 3^p + N_2\, 2^p + N_1\, 1^p}{N}} \right\rceil$$

### 6.3.2 The Real-Time Monitoring and Risk Assessment module in the Fire pilot

During the emergency phase of a forest fire, the Real-Time Monitoring and Risk Assessment module enables crisis managers to monitor the ongoing crisis as well as to assess its severity level dynamically based on the information obtained directly from the field. It receives real-time weather observations, such as air temperature, humidity, precipitation, and wind speed/direction and along with the real-time information from the multimedia (image and video) analysis components, estimates the ongoing crisis's severity level and it updates the crisis managers about the crisis severity evolvement.

The proposed algorithm for risk assessment relies heavily on the incoming information from first responders and citizens. They can report an incident by sending it to the system via mobile application including the captured images and videos. The incoming incidents are grouped and forwarded to the Decision Fusion module in order to be furtherly analysed and fused. The region of Valencia has divided into 4 fire zones of interest (Figure 16). For the estimation of the risk/severity level in the region of Valencia, a hierarchical bottom-up approach has been adopted. When a new incident appears into the beAWARE ecosystem, it is clustered into one of the existing groups. The *Real-Time Monitoring and Risk Assessment* module estimates the severity level of the modified group/cluster. Then, the severity level of the whole fire zone, that the cluster belongs to, is updated. After that, the overall fire risk assessment/severity level in the region of Valencia will update automatically.


Figure 16: Fire Zones in Valencia Region

Briefly the steps of the risk assessment algorithm are the following:

**Step 1.** **Data Acquisition**: includes the processes to retrieve the outcomes of the multimedia analysis from the corresponding beAWARE modules. It should be mentioned, that when a citizen or a first responder sends an image or video to the system, then it is characterized as an incident. It potentially belongs to a cluster, so it is grouped in the corresponding cluster. Then, it is analysed from the corresponding components and the outcomes are forwarded to the system.

**Step 2.** **Decision Fusion:** estimates the risk assessment by the exploitation of the received information from the multimedia analytical tools. In this step, the severity level of the incident and/or the cluster of incidents are updated. Then, the severity level of the whole fire zone in which the incident or cluster belongs to is updated correspondingly.

**Step 3.** **Store** the incident report and results of the analysis to the MongoDB database.

**Step 4.** **Create** the appropriate messages for both PSAP and dashboard in order to update the status of each incident and the status of the fire zone (gauge and traffic light plots).

In more details, the decision fusion algorithm classifies the participants (objects/targets) in images and video frames into two classes:

- class_1 = ["bicycle", "bus", "car", "motorcycle", "truck"]
- class_2 = ["cat", "dog", "human", "wheelchair user"]

If the EmC module of the Image/Video Analysis component detects fire or smoke in the incident, then for each one of the incident's participant (target), it does the following:

a) Determine the class the participant belongs to.
b) Determine the weight of the participant in terms of the risk and confidence metrics along with the number of the particular participant in the media (image or video frame). The indicators risk and confidence are the qualitative outcome of the Image/Video analysis component as mentioned in the section 2.2.3 . Thus, the weight is set by employing the following table:

Table 14: Weight table for each class of participants depending on the image/video analysis results

| **Confidence** / **Risk** | **Low** | **Medium** | **High** |
|---|---|---|---|
| **Low** | $w_{cl1} = 0.25$<br>$w_{cl2} = 0.35$ | $w_{cl1} = 0.5$<br>$w_{cl2} = 0.6$ | $w_{cl1} = 0.6$<br>$w_{cl2} = 0.7$ |
| **Medium** | $w_{cl1} = 0.5$<br>$w_{cl2} = 0.6$ | $w_{cl1} = 0.6$<br>$w_{cl2} = 0.7$ | $w_{cl1} = 0.7$<br>$w_{cl2} = 0.8$ |
| **High** | $w_{cl1} = 0.6$<br>$w_{cl2} = 0.7$ | $w_{cl1} = 0.7$<br>$w_{cl2} = 0.8$ | $w_{cl1} = 0.85$<br>$w_{cl2} = 0.95$ |

For each participant, its risk/severity ($R_{part}$) is estimated by the formula:

$$R_{part} = \frac{1}{num\_detections} * \sum_{i=1}^{num\_detections} w_{cl.,i}$$

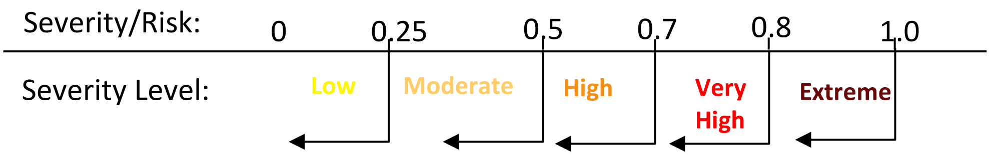
where *num_detections* denotes the number of objects that participate in the same incident. For example, in one image (incident) six humans could participate (detected by the Image Analysis module). Thus, the participant type is "human" and the number of detections is equal to six (6). For each detection, different analysis results in terms of risk and confidence could appear.

c) Use the linear weighted fusion approach in order to estimate the risk/severity of the incident or the cluster of incidents.

$$R_{cluster} = \frac{1}{num\_part} * \sum_{i=1}^{num\_part} w_i * R_{part,i}$$

Here, we can consider that each one of the participants has equal contribution to the estimation of the risk. Thus, the $w_i = 1$.

d) Define the severity level of the cluster of incidents by using the scale below:




e) Define the severity level of the whole of fire zone based on the severity levels of the clusters and incidents in the specific fire zone. The majority voting fusion algorithm is utilised for this purpose.

# 7 Evaluation

## 7.1 Visual analysis

So far, we have evaluated most of the visual analysis modules and functions in the technical evaluation report of D7.6. We will provide some quantitative and qualitative evaluation reports here for every function that was integrated after D7.6. Those are:

1) Final version of the EmC which includes smoke detection
2) Final version of the ObD which included wheelchair user detection and animal detection

### 7.1.1 EmC final version

We have trained the EmC with over 20000 images taken from public datasets and beAWARE data. The mean accuracy of the classifier on those data is 0.98%. Figure 17 shows the confusion matrix for the emergency classification task and Figure 18 the normalized version. As we can see there are very good accuracy rates in general. Most of the errors are false positive cases, i.e. images where no emergency is taking place but are classified as flood in 81 cases, fire in 51 cases and smoke in 13 cases. Some false negatives also appear. Interestingly, there is very little confusion between the emergency classes. There are just 31 fire instances that have been misclassified as smoke and only 11 smoke instances misclassified as fire. Moreover, flood is almost never confused with fire or smoke instances. This means that even though there is one unified EmC model for all emergency cases, the visual analysis will rarely confuse emergency events.

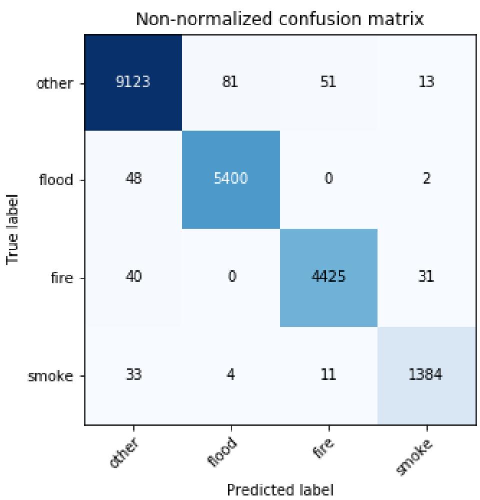



Figure 17: Non-Normalized confusion matrix of the EmC

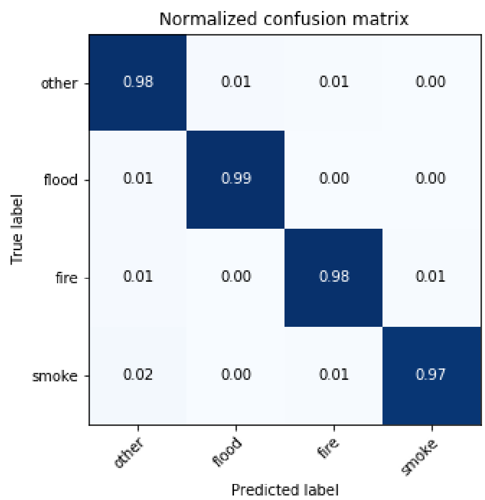



Figure 18: Normalized confusion matrix of the EmC


Figure 19: Flood false positive

Figure 20: Flood false positive

Figure 21: Fire false positive

Figure 22: Fire false positive

Figure 23: Smoke false positive

Figure 24: Smoke false positive

We examine some false positive cases in Figure 19-Figure 24. As shown in the figures, the flood false positives are images where water bodies are found but the water level is low. However, even as false positives the thematic content of the images are not entirely foreign to the context of a flood event which is a good indicator of the quality of support the beAWARE system can provide to the authorities even when a wrong prediction happens. Some hard examples are also shown for the fire use case and the smoke use case. The fire texture can sometimes be found in fireworks and smoke is sometimes found in areas with high humidity, like clouds or fog, which can resemble smoke textures.

### 7.1.2 Object detection final version

We will provide some qualitative examples of our final object detector in order to showcase its extended applicability. Figure 25 shows a dog that has been detected close to a fire texture. The detector provides accurate bounding box coordinates even in the presence of abnormal illumination due to the flames. In Figure 26 a wheelchair user has been detected crossing the street. Figure 27, Figure 28 and Figure 30 show the detected pedestrians that are walking inside the flooded streets. In those images, the detector's robustness in viewpoint and scale variation is showcased.

Figure 25: A dog detected in a fire incident

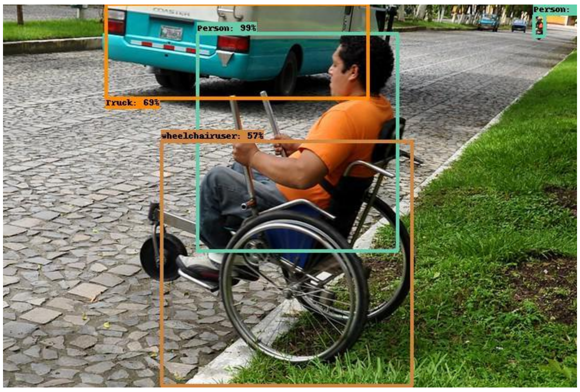


Figure 26: A detected wheelchair user

Figure 27: Pedestrians detected in a flooded area

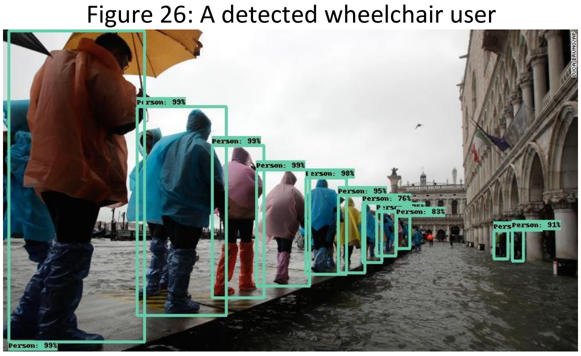


Figure 28: Pedestrians detected crossing a flooded area

Figure 29: People and vehicles detected close by a fire outbreak

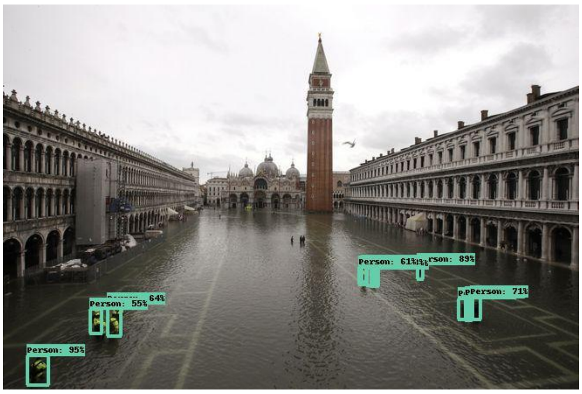


Figure 30: Pedestrians detected successfully in small scale

### 7.1.3 Visual River Sensing

Initial evaluation results of the VRS module, as it was developed in the second version of the platform, were presented in D7.6. During the third development circle of the project, the VRS algorithm has been enhanced by averaging the estimated results over multiple frames, instead of just using a single frame, as it was in the initial implementation. Additionally, the resolution of the streamed video was increased to 720p (1280x720) from the initial 640x340, in order to further improve accuracy. For the evaluation of the module, the same annotated dataset was used as in D7.6, which was a series of video captures from 2016, during a flooding event. The annotations were water level measurements from a sensor installed in the middle of Angeli Bridge. The following chart (Figure 31) presents the results of the water level estimation algorithm on the evaluation dataset, for the two different versions of the module. From the chart, it can be observed that the aforementioned enhancements on the code, improved the estimation accuracy, by reducing the estimation error in some specific cases.

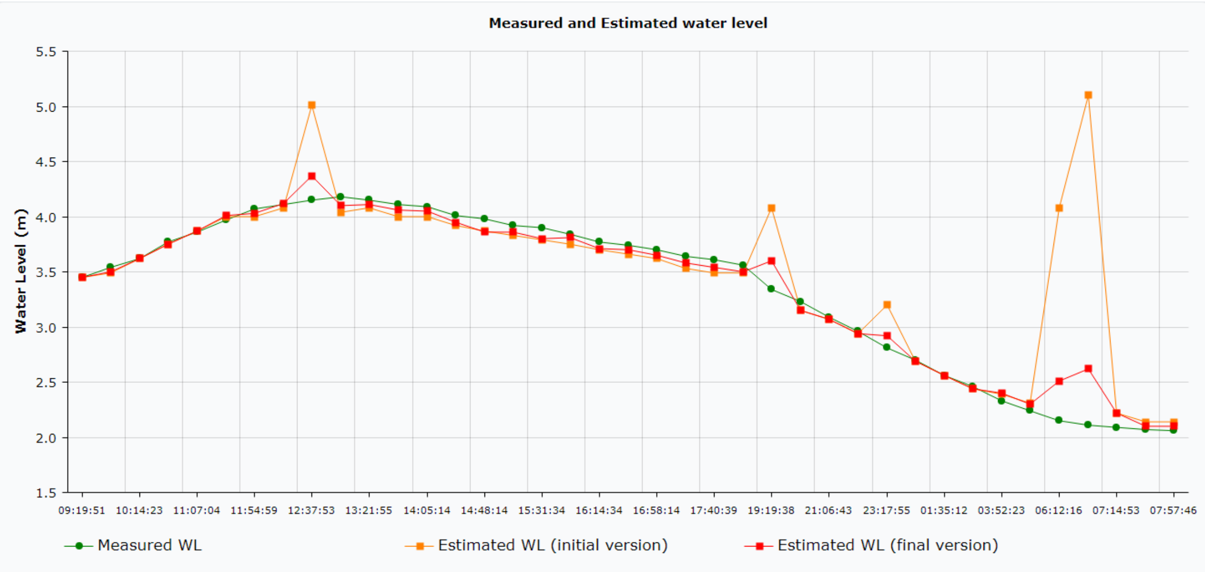



Figure 31: Comparison of water level values measured from the sensor (green) and estimated by the initial version of VRS (orange) and by the final version (red)

On the other hand, the improvement in accuracy did not seem to have a substantial effect on time efficiency, since the average analysis time on 10 consecutive runs was slightly increased from 3.28 seconds to 4.21 seconds. Thus, the average total analysis time, including the 10 seconds of the streaming part, is 14.21 seconds.

## 7.2 Drone analysis

In deliverable D7.6 the object detection functionality of the Drones Analysis component was evaluated by using a detection model that was specifically trained for the detection of a dummy that was used to fulfill the needs of the Flood Pilot. At the final version of the platform, the Drones Analysis uses a detection model, based on a faster R-CNN architecture, able to detect people and vehicles and additionally an image classification model able to detect ‘fire’, ‘smoke’ and ‘flood’. In order to evaluate the object detection model, video frames from 8 video sequences were extracted, containing people, bicycles and cars. Extracted frames were manually annotated in order to create ground-truth images. Subsequently, the Average Precision (AP)[21] was calculated for each object class by estimating the area under the curve of the precision-recall curve. Results are presented in Table 15. The Mean Average Precision (mAP) is calculated by taking the mean AP over all classes. It should be noted that the 'bicycle' class significantly affects mAP, because of its low precision, however this is a very limited class, with around 5% of the total object instances. Regarding the image classification model that is used for detecting fire, smoke and flood, since it is the same model as the one used by EmC, its evaluation already was presented in the section 7.1.1 .

Table 15: Object Detection performance of the Drones Analysis component

| | Class | | | mAP% |
|---|---|---|---|---|
| | Car | Bicycle | Person | |
| **Average Precision** | **64.12%** | **20.15%** | **62.19%** | **48.82%** |

## 7.3 Audio analysis

During the third development period, the focus was mainly on the Spanish model, by enriching the dictionary and updating the language model (LM). Additional improvements have to do with the format of the LM, which was affecting the recognition result and with audio quality produced by the audio converter that is used from the Mobile App. In order to visualize the effect of these modifications, the following sample text was used:

Original Text:

"Me llamo Pedro y hoy quiero hablar del parque que hay junto a mi casa. Yo me divierto todos los días en el parque. Allí veo las palomas comiendo y bebiendo agua. También veo pájaros de colores en los árboles. Yo voy al parque a las cinco de la tarde, cuando termino los deberes de la escuela. Allí veo a mi amigo Juan y a mi amigo Luis. Con ellos juego al escondite y a otros juegos muy entretenidos. Luis se va más temprano del parque porque tiene que ir a la escuela de música a aprender a tocar el piano."

This text was recorded at Fs=16kHz and BitRate=256kbps. The recorded audio was initially subjected to speech recognition by using the Spanish model as it was at the second version of the platform. The audio was sent through the Mobile App to ASR for analysis and the result is presented below. The words that were misrecognized are highlighted:

[21] https://github.com/rafaelpadilla/Object-Detection-Metrics

**ASR result with second version model** (*Comparison with original text):*

"me llamo pedro y hoy quiero hablar del **paquete ahí con tony** caza. yo me **despierto** todos **os di as** en el **paquete**. allí **hebreo** las **pa lomas** comiendo **deberían de** agua. también veo pájaros de colores **solos** árboles. **y** voy **a partir** a las cinco **dela** tarde cuando **terminamos de pares** de **las cuerdas**. allí **y** veo a mi amigo **cual** y a mi amigo luis. con ellos **cueva les con pete** y a otros **cuerpos muy entre tenido as. louise va más d em.**"

Subsequently, ASR was performed once again on the same audio, through the Mobile App, by using the text format of the LM, instead of the binary format. From the following result, it can be observed that, even though the binary format is commonly used to reduce size and increase performance, it has a negative effect on the accuracy. This can be observed by the fallen percentage of misrecognized words and thus, the model was reverted back to its text format.

**ASR result with non-binary LM** (Comparison with original text):

"me llamo pedro y hoy quiero hablar **de paquete ahí con toda** mi casa. yo me **despierto** todos los días en el parque. allí **hebreo** las **pa lomas** comiendo **deberían de** agua. también veo pájaros de colores en los árboles. **y** voy al parque a las cinco de la tarde cuando terminó los deberes de la escuela. allí **y** veo a mi amigo **cual** y a mi amigo **lois**. con ellos **cueva les con pete** y a otros **cuerpos** muy **entre tenido as. louise** va más temprano **de parte** porque tiene que ir a la escuela de música **(a)** aprender a tocar el piano."

Finally, the audio was processed once again by using the text format of the LM, after fixing the issue with the audio converter of the Mobile App. Specifically the audio conversion quality was increased from 16kbps to 256kbps. The produced transcription is presented below. The underlined text is the misrecognized text (compared to the original text), whereas the text that is colored green represends the improvement compared to the transcription produced without the fix on the audio converter. From the following result, it can be observed that there is an additional small improvement on the recognition accuracy.

**ASR result with non-binary LM** (Comparison with original text (underlined) and with the text produced without the fix on the converter (green)):

"me llamo pedro y hoy quiero hablar **del** **paquete el culto** **a** mi casa. yo me **despierto** todos los días en el parque. allí **hebreo** las **pa lomas** comiendo **deberían de** agua. también veo pájaros de colores en los árboles. **y** voy al parque a las cinco de la tarde cuando terminó los deberes de la escuela. allí **y** veo a mi amigo **cual** y a mi amigo **luis**. con ellos **cueva les con pete** y a otros **cuerpos** muy **entre tenido as. louise** va más temprano **de parte** porque tiene que ir a la escuela de música **a** aprender a tocar el piano."

A final evaluation step was to check if the enrichment of the Spanish dictionary and the LM, with text from emergency-related corpus, had a negative effect on the recognition of random (non-case specific) text, such as the one used above. For this purpose, the same audio was subjected to speech recognition by using first, the initial dictionary along with the text format of the initial LM and then, by using the enriched dictionary with the text format of the enriched LM. From the results below, it can be observed that the two models produced exactly the same errors compared to the original text, which means that there was no influence of the model update on irrelevant text.

Result with initial dictionary and LM (comparison with original text):

"me llamo pedro y hoy quiero hablar del **paquete el culto** a mi casa. yo me **despierto** todos los días en el parque. allí **hebreo** las **pa lomas** comiendo **deberían de** agua. también veo pájaros de colores en los árboles. **y** voy al parque a las cinco de la tarde cuando terminó los deberes de la escuela. allí **y** veo a mi amigo **cual** y a mi amigo lois. con ellos **cueva les con pete** y a otros **cuerpos** muy **entre tenido as. louise** va más temprano de **parte** porque tiene que ir a la escuela de música a aprender a tocar el piano."

Result with enriched dictionary and LM:

"me llamo pedro y hoy quiero hablar del **paquete el culto** a mi casa. yo me **despierto** todos los días en el parque. allí **hebreo** las **pa lomas** comiendo **deberían de** agua. también veo pájaros de colores en los árboles. **y** voy al parque a las cinco de la tarde cuando terminó los deberes de la escuela. allí **y** veo a mi amigo **cual** y a mi amigo lois. con ellos **cueva les con pete** y a otros **cuerpos** muy **entre tenido as. louise** va más temprano de **parte** porque tiene que ir a la escuela de música a aprender a tocar el piano."

For a quantitative evaluation, a short set of 20 Spanish phrases was recorded from native Spanish speakers (PLV members) at 16kHz and 256kbps, containing context related to the Fire scenario. Recorded audios were sent to ASR through the Mobile App and were subjected to analysis. The measure used for the evaluation was percent Word Error Rate (%WER), which is defined as:

$$\%WER = 100 * \frac{S + D + I}{N}$$

where

- $S$ is the number of word substitutions,
- $D$ is the number of deletions,
- $I$ is the number of insertions and
- $N$ is the number of words in the reference text.

The average %WER of the updated model was 13.03%, whereas average %WER of the initial model on the same set was 17.02%. However, it should be noted that there is a significant difference between ASR results on audio coming from the Mobile App and the call center, since it was not possible to increase the recording audio quality of the call center recorder, as it was mentioned in Section 4.

## 7.4 Text analysis

In D7.6 we reported a comparative domain-independent evaluation of the UD and PennTreBank versions of the linguistic analysis component. Its results were the basis for using UD in the final version describe in this component. We have conducted further extrinsic evaluations of the UD-based linguistic resources and models using NLG as a downstream task. These are reported in D5.3.

In this deliverable we focus on the knowledge extraction part of text analysis, and present a quantitative evaluation using materials pertinent to the 3rd pilot of the final versions of the geolocation, disambiguation and concept extraction components. All these components have been largely developed within the scope of beAWARE. We also describe and present the results of a qualitative evaluation of the final output of the TA module. As we already did for D7.6, we use the same set of tweets across all these evaluations. This time we use a set of 40 tweets, 20 of them are in Spanish and the other 20 are their translations into English. The first 11 tweets in each language correspond to actual inputs used in the 3rd pilot, while the remaining 9 were added to widen the range of contents and alternative wordings for incidents, states, locations and impacted objects. Table 16 shows the English translations of the tweets.

Table 16: Set of tweets used to evaluate the text analysis module

| | |
|---|---|
| **Text 1** | Fire risk in El Saler, high temperatures and strong wind, be aware. |
| **Text 2** | It is so hot and windy in Valencia. A wildfire occurring would be hard to handle. |
| **Text 3** | It is so hot in the Albufera Natural Park, so be careful, there is a risk of fire. |
| **Text 4** | There is an extreme risk of fire in La Devesa de l'Albufera due to high temperatures. |
| **Text 5** | It is so windy and hot in la Devesa, so the risk of fire is very high. |
| **Text 6** | The Albufera Natural Park is in risk of fire due to the high temperatures and strong wind. |
| **Text 7** | Smoke is seen in la Devesa, there is a fire. |
| **Text 8** | Right now there is a fire in el Saler, please be careful. |
| **Text 9** | From the CV-500 highway we can see a cloud of smoke in the Tallafoc de la Rambla area. |
| **Text 10** | From the Racó de l'Olla bird observatory I can see a fire northwards. Please, it needs to be extinguished! |
| **Text 11** | Fire is moving towards the instituto del Saler. There's students inside. |
| **Text 12** | Rain in el Saler and fire is now contained. |
| **Text 13** | Raindrops are now falling in Valencia, but the fire is still advancing. |

| | |
|---|---|
| **Text 14** | Wind is weaker now in la Devesa and it looks as if it might rain. This is causing the fire to retreat. |
| **Text 15** | It is raining in the Playa dels Ferros, but wind is still strong. |
| **Text 16** | Rain and weaker wind are causing the fire to retreat in the Albufera area. |
| **Text 17** | Wind has stopped and the fire has been extinguished in the Albufera Natural Park. |
| **Text 18** | Plenty of smoke in the CV-500, cars are stopped. |
| **Text 19** | It looks like it may rain in Valencia soon. |
| **Text 20** | Fire is spreading towards the secondary school in El Saler, people there need to be evacuated! |

### 7.4.1 Disambiguation

The evaluation of the disambiguation component for the 3rd pilot replicates the approach adopted for the 2nd pilot. First, the set of English tweets and their Spanish counterparts were manually annotated with meanings in BabelNet 4.0.1 following the guidelines included in the appendices of D7.6. We then calculated precision, recall and F1 by comparing system and manually annotated gold meanings. We use a subset of the baselines deescribed in D7.6, random disambiguation, BabelNet First Sense (BFS), and Bias.

Table 17: F1 results for the Spanish tweets

| | Nouns | Verbs | Adjectives | Adverbs | All |
|---|---|---|---|---|---|
| | 48 | 13 | 11 | 11 | 80 |
| Random | 0.32 | 0.24 | 0.38 | 0.47 | 0.33 |
| BFS | 0.58 | 0.46 | 0.96 | 0.71 | 0.64 |
| Bias | 0.80 | 0.28 | 0.86 | 0.59 | 0.74 |
| Rank | 0.80 | 0.19 | 0.86 | 0.59 | 0.71 |

Table 18: F1 results for the English tweets

| | Nouns | Verbs | Adjectives | Adverbs | All |
|---|---|---|---|---|---|
| | 51 | 17 | 10 | 11 | 84 |
| Random | 0.34 | 0.20 | 0.64 | 0.30 | 0.30 |
| BFS | 0.30 | 0.30 | 0.95 | 0.50 | 0.40 |
| Bias | 0.62 | 0.20 | 0.95 | 0.40 | 0.55 |
| Rank | 0.64 | 0.25 | 0.95 | 0.40 | 0.56 |

Table 17 and Table 18 show the results of our evaluation using precision (P), recall (R) and F-score (F1) metrics for Spanish and English tweets, respectively. Results are also shown split down by POS. As in the evaluation for the 2nd pilot, the Bias and Rank systems perform better than the baselines for nominal expressions, but this time the complete implementation has a significant lead over the limited version. Also similar to what was shown for the 2nd pilot, our component struggles to match the performance of the BFS baseline when applied to other grammatical categories.

### 7.4.2 Geolocation

Evaluation of the geolocation module was performed on a dataset specifically designed for the project. The dataset consists of artificial tweets about a fire scenario manually written in Spanish by a professional linguist. Each tweet contains a location within the Albufera natural park in Valencia. The possible locations are conditioned by appearing in either Open Street Maps or GeoNames. The maximum length restriction of Twitter messages was observed. References to the locations can use full names in the geodatabases or parts of it, e.g. "Perelló" instead of "El Perelló". The addresses are written in Spanish and not in Catalan e.g. "calle Gavines", "Parque de la Albufera", etc. Each tweet was annotated with the URLs of the location in Open Street Maps and GeoNames.

Geodata maps sometimes have several nearby instances corresponding to the same entity in a text. For such entities, firstly all possible coordinates with all possible normalized names were added as separate ground truth entities. Secondly, to have a single ground truth example per entity the test set was finalized as follows. Coordinates of each test entity were surrounded by a rectangle with side length equal to 1km. Then test entities that shared a common area were grouped into one positive example and the rectangle with the minimum square that covered all subareas was chosen to set the boundaries for the joint entity.

Some entities in texts of the test set cannot be localized correctly in their initial form due to the provided possibility to use only parts of the full names and the necessity to write an address in Spanish even if it is absent in Geodata bases in this language. For example, "*entrada al Golf*" which refers to "*Golf Parador de El Saler*", "*camino del Port*" with the correct name "*Camí del Port*", and "*club de piragüismo Silla*" corresponding to a ground truth entity "*Club esportiu piraguisme Silla*" cannot be found in search indexes developed within the project nor with the search services of the online versions of geolocation tools. Therefore, we first assessed what values of measures might be obtained in the assumption that all the entities are detected correctly. In this case, the presicion is equal to 0.67, recall – to 0.77, and $F_1$-score – to 0.72. These values should be treated as maximum possible values that might be achieved on this dataset. We used upper bound of $F_1$-score for normalization of the measures for models under evaluation. Since it is often enough to correctly localize only one object in a sentence to get the correct coordinates of an event, we also calculated a share of texts with at least one correct geolocation found.

To contrast our geolocation model using SpaCy as a core algorithm for location name detection, we selected two robust named entity recognizers as baselines: NER Tagger (Lample et al., 2016) and CoreNLP NER (Finkel et al., 2005). While the NER Tagger is based on bidirectional LSTMs and Conditional Random Fields (CRFs) that rely on character-based word representations and unsupervised word representations learned from unannotated corpora, the Stanford CoreNLP NER is an implementation of linear chain Conditional Random Field (CRF) sequence models. We ran two versions of Stanford CoreNLP trained on different corpora (ancora, kpb.ancora).

Table 19: Performance indicators for geolocation

| CoreNLP NER algorithm | P | R | $F_1$-score | Share of texts with at least one correct geolocation found |
|---|---|---|---|---|
| NER Tagger (Lample et al., 2016) | 0.55 | 0.63 | 0.59 | 0.6 |
| StanfordCoreNLP (ancora) | 0.46 | 0.54 | 0.5 | 0.53 |
| StanfordCoreNLP (kpb.ancora) | 0.57 | 0.6 | 0.58 | 0.57 |
| SpaCy | 0.49 | 0.69 | 0.58 | 0.64 |

Table 19 shows that the core algorithm chosen as a basis of our geolocation module outperforms other promising NER algorithms by 10% in terms of recall with the second-best $F_1$-score that is less than a percentage point lower than the best-reached value. In terms of correctly geolocated events, our model also reaches the highest accuracy.

### 7.4.3 Concept extraction

In order to evaluate the module, the set of tweets in Table 16 was manually annotated in compliance with the same guidelines previously used for the evaluation within the 2nd pilot:

- annotate only noun phrases as concepts;
- all nouns should be included into annotation; a noun in a particular position in a sentence should appear only in one concept;
- noun phrase should be treated as a concept if it represents a single piece of knowledge and is closer to semantically undividable unit rather than to compound phrase in a given context; if there are several embedded concepts, the one with the largest span should be annotated;
- rarely occurred or novel multi-word noun phrase that potentially might become a concept should be annotated as a single concept only if they form a proper name.

The obtained annotation which included 100 unique concepts per language was used as ground truth for evaluating the results of the automatic concept extraction.

DBPedia Spotlight (Daiber et al. 2013) that is the most prominent dictionary lookup implementation was chosen as a strong baseline as it outperforms rule-based concept extractors and most of the state-of-the-art machine-learning-based approaches including neural sequence labeling approaches. It matches and links identified nominal chunks with DBpedia entries (Bizer et al. 2009), based on the Apache OpenNLP[22] models for phrase chunking and named entity recognition. Given the large coverage of DBpedia (6.6M entities, 13 billion RDF triples)[23], the performance of DBpedia Spotlight is rather competitive.

The values of key measures for English and Spanish are presented in

[22] https://opennlp.apache.org/

[23] https://wiki.dbpedia.org/develop/datasets/dbpedia-version-2016-10

Table 20 and Table 21 correspondingly. The obtained results show that the module outperforms the strong baseline by 5% and provides both precision and recall high enough so that the proposed model is suitable for use within the crisis management domain.

Table 20: Performance indicators for concept extraction for English

| | Precision | Recall | $F_1$-score |
|---|---|---|---|
| DBPedia Spotlight | 0.74 | 0.72 | 0.73 |
| Our model | 0.78 | 0.73 | 0.76 |

Table 21: Performance indicators for concept extraction for Spanish

| | Precision | Recall | $F_1$-score |
|---|---|---|---|
| DBPedia Spotlight | 0.71 | 0.64 | 0.67 |
| Our model | 0.71 | 0.72 | 0.71 |

### 7.4.4 Concept and relation extraction

In our qualitative evaluation using the 2nd pilot tweets we focused on the extracted incidents, their participants and locations. We repeat the same approach for the 3rd pilot, as both disambiguated meanings locations have been evaluated separately. This time, however, we extended the evaluation to also include states associated with the incidents. We manually analyzed the pilot texts to determine the expected outputs for each of the tweets in Table 16. These expected results are listed in Table 22, along with the actual outputs for the English and Spanish texts. Errors in the outputs are marked in red.

In line with the results reported in D3.3 for the basic version of the text analysis module, the advanced version can correctly detect the incident or event communicated in the tweets, even when there are multiple of them. Both incidents and locations are correctly detected in all cases except for a missing wind incident in the second English tweet and a location in the last English twet. States, on the other hand, have a more uneven rate of detection, with 10 out 16 states being correctly detected in the English tweets, and the same number for the Spanish tweets. In addition, three incidents were attributed with incorrect states in the first three Spanish tweets.

Table 22: Expected and actual outputs of the TA module

| | Expected output | English output | Spanish output |
|---|---|---|---|
| **Text 1** | Fire: Risk, Location<br>Wind: Greater, Location<br>Heat: Greater, Location | Fire: Risk, Location<br>Wind: Greater, Location<br>Heat: Greater, Location | Fire: Risk, Location<br>Wind: Greater, Location<br>Heat: Greater, Location, **Risk** |
| **Text 2** | Fire: Risk, Location<br>Wind: Greater, Location<br>Heat: Greater, Location | Fire: Risk, Location<br>**Wind: Greater, Location**<br>Heat: **Greater**, Location | Fire: Risk, Location<br>Wind: **Greater**, Location, **Risk**<br>Heat: **Greater**, Location |
| **Text 3** | Fire: Risk, Location<br>Heat: Greater, Location | Fire: Risk, Location<br>Heat: **Greater**, Location | Fire: Risk, Location<br>Heat: **Greater**, Location |

| | | | |
|---|---|---|---|
| **Text 4** | FIRE: RISK, LOCATION<br>HEAT: GREATER, LOCATION | FIRE: RISK, LOCATION<br>HEAT: GREATER, LOCATION | FIRE: **RISK**, LOCATION, **GREATER**<br>HEAT: GREATER, LOCATION |
| **Text 5** | FIRE: RISK, LOCATION<br>WIND: GREATER, LOCATION<br>HEAT: GREATER, LOCATION | FIRE: RISK, LOCATION<br>WIND: **GREATER**, LOCATION<br>HEAT: **GREATER**, LOCATION | FIRE: RISK, LOCATION<br>WIND: **GREATER**, LOCATION<br>HEAT: **GREATER**, LOCATION |
| **Text 6** | FIRE: RISK, LOCATION<br>WIND: GREATER, LOCATION<br>HEAT: GREATER, LOCATION | FIRE: RISK, LOCATION<br>WIND: GREATER, LOCATION<br>HEAT: GREATER, LOCATION | FIRE: RISK, LOCATION<br>WIND: GREATER, LOCATION<br>HEAT: GREATER, LOCATION |
| **Text 7** | FIRE: LOCATION<br>SMOKE: LOCATION | FIRE: LOCATION<br>SMOKE: LOCATION | FIRE: LOCATION<br>SMOKE: LOCATION |
| **Text 8** | FIRE: LOCATION | FIRE: LOCATION | FIRE: LOCATION |
| **Text 9** | SMOKE: LOCATION1, LOCATION2 | SMOKE: LOCATION1, LOCATION2 | SMOKE: LOCATION1, LOCATION2 |
| **Text 10** | FIRE: LOCATION | FIRE: **LOCATION** | FIRE: LOCATION |
| **Text 11** | FIRE: LOCATION, GREATER, PEOPLE | FIRE: LOCATION, GREATER,PEOPLE | FIRE: LOCATION, GREATER, PEOPLE |
| **Text 12** | RAIN: LOCATION<br>FIRE: LOCATION, LESSER | RAIN: LOCATION<br>FIRE: LOCATION, LESSER | RAIN: LOCATION<br>FIRE: LOCATION, **LESSER** |
| **Text 13** | RAIN: LOCATION<br>FIRE: LOCATION, GREATER | RAIN: LOCATION<br>FIRE: LOCATION, GREATER | RAIN: LOCATION<br>FIRE: LOCATION, **GREATER** |
| **Text 14** | WIND: LOCATION, LESSER<br>RAIN: LOCATION. RISK<br>FIRE: LOCATION, LESSER | WIND: LOCATION, LESSER<br>RAIN: LOCATION. RISK<br>FIRE: LESSER | WIND: LOCATION, LESSER<br>RAIN: LOCATION. RISK<br>FIRE: LESSER |
| **Text 15** | RAIN: LOCATION<br>WIND: LOCATION, GREATER | RAIN: **LOCATION**<br>WIND: **LOCATION**, GREATER | RAIN: LOCATION<br>WIND: LOCATION, GREATER |
| **Text 16** | RAIN: LOCATION<br>WIND: LOCATION, LESSER<br>FIRE: LOCATION, LESSER | RAIN: LOCATION<br>WIND: LOCATION, LESSER<br>FIRE: LOCATION, LESSER | RAIN: LOCATION<br>WIND: LOCATION, **LESSER**<br>FIRE: LOCATION, LESSER |
| **Text 17** | WIND: LOCATION, STOP<br>FIRE: LOCATION, STOP | WIND: LOCATION, STOP<br>FIRE: LOCATION, STOP | WIND: LOCATION, STOP<br>FIRE: LOCATION, STOP |
| **Text 18** | SMOKE: LOCATION, GREATER, VEHICLE | SMOKE: LOCATION, GREATER, VEHICLE | SMOKE: LOCATION, GREATER, VEHICLE |
| **Text 19** | RAIN: LOCATION, RISK | RAIN: LOCATION, RISK | RAIN: LOCATION, **RISK** |
| **Text 20** | FIRE: LOCATION, GREATER, PEOPLE | FIRE: LOCATION, GREATER, PEOPLE | FIRE: LOCATION, GREATER, PEOPLE |

Many errors are caused by problems in the linguistic analysis, either because wrong PoS prevents successful disambiguation -e.g. 'windy' marked as a noun prevents finding the right candidates for disambiguation and this results in a failed mapping to the *Wind* ontology class- or because parsing errors result in words indicating a state not governing the right incident – e.g. 'Riesgo' being incorrectly attached to 'temperaturas' causes relation extraction to produce a state *Risk* for *Heat*. In other cases, the disambiguation component fails to produce the right result. This is particularly true for adverbs and adjectives whose meaning is not related to the domain, i.e. those indicating states: 'weaker', 'high'. Finally, there are three tweets in which the geolocation component fails to match names of places with the right entry in the datasets it uses. This is caused by the NER tools failing to detect certain names, e.g. 'Playa dels ferros' in English, or due to the fact that such entry might not exist at all in the database.

Despite these limitations, it is thanks to the improved disambiguation and geolocation components that the detection of states can be addressed in the final version of the text analysis module. This is crucial to detect hypothetical mentions to incidents, such as a risk of fire- in tweets that are otherwise relevant. In addition, the lack of precision of the text analysis module is compensated by the clustering functionality of the crisis classification. Incidents are only shown to users in the PSAP if enough evidence is gathered in a certain geographical area including the analysis of texts as well as other inputs. This means that even if a tweet indicating a hypothetical event is wrongly analyzed as indicating a factual event, the analysis result on its own is unlikely to cause an incident to be shown up in the PSAP.

## 7.5 **Crisis Classification**

Although in the DoA the Crisis Classification component was not foreseen to be described in this deliverable, for the sake of completeness, we include in this report its new functionalities, which have been designed and implemented during the 2nd and final prototype.

The evaluation for the flood risk assessment algorithm was described in detail at the 2nd Technical Evaluation Report (Deliverable D7.6) which reflects the second release version of the beAWARE system compiled in M24. In that document, the performance of each one of the Crisis Classification modules, in terms of the amount of data (forecasts, real-time observations) to process, execution time and accuracy of the results was examined. The evaluation mainly focused on testing the new functionalities that integrated into the 2nd prototype, namely the risk maps and the novel risk assessment algorithm for flood crisis events.

The evaluation of the functionalities that implemented and encapsulated in the final version of the component will be reported in the forthcoming deliverable D7.9, which will reflect the final technical evaluation of the beAWARE system. In that deliverable, the final version of the Crisis Classification component will be evaluated and the results of the fire risk assessment algorithm, as it was tested during the 3rd pilot in Valencia, will be presented and commented.

# 8 Conclusions

This deliverable reports on the advanced methods for T3.1, T3.2 and T3.3, and the final versions of the Image, Video, Audio, Drones and Text analysis modules, as well as the Crisis Classification module. It also reports technical evaluations for components belonging to each of these modules. The contents of this deliverable describe the work in the period M24-36 towards MS4 (Final protoype) of WP3 and elaborate on the previous reports D3.3 for MS2 (First prototype) and D7.6 for MS3 (Second prototype).

From the advances and the results of the various evaluations reported above, we draw the following conclusions:

1. Image and Video: We have confirmed the applicability of the visual analysis advanced techniques by evaluating its separate modules. The Emergency Classification evaluation was performed in a large-scale database that was gathered from various sources especially for the final version. The accuracy rates show a very good performance, with minor false positive cases which were further studied. The false positives appear to be hard samples due to inter-class texture similarity in most cases. The extended Object Detection was evaluated using qualitative examples that show the robustness of the detector to scale, illumination or occlusion variations. Additionally, since the submission of D3.3 a module called Visual River Sensing was added. This module, at the second version of the platform was able to stream video from static cameras, analyze it in order to estimate the water level, create relevant alerts and forward analyzed footage to the respective component for further traffic analysis. At the last version of the platform the water level estimation has been enhanced by averaging the result over more frames and by extracting video of higher resolution, resulting this way in more accurate estimation.
2. Drone analysis: The Drone Analysis module was the last addition to the platform. In the second prototype, this module was able to analyze sequences of images in order to perform object detection and tracking for the detection of people and vehicles and image segmentation for the detection of flooded areas. In the final version of beAWARE, the module has been enhanced in order to analyze both image and video sequences, the image segmentation model has been replaced by an image classification model able to detect flood, fire and smoke and an additional task has been defined, in order to support evacuation missions and provide evacuation reports.
3. Audio Analysis: Automatic Speech Recognition has also been enhanced by suitably updating the language models, acoustic models and dictionaries. A call center solution has also been added at the second version of the platform in order to receive and analyze emergency calls. Technical issues have been resolved also at the final version in order to improve the accuracy of recognition.
4. Text Analysis: As indicated in D3.3, our efforts in text analysis have focused on delivering improved versions of disambiguation and geolocation components, as these components are crucial to crisis classification and to obtain better coverage in concept and relation extraction. The geolocation component performs remarkably well considering its relative simplicity. Disambiguation work particularly well at detecting incidents and impacted objects over a wide range of words and phrases. In addition, the relation and conceptual relation extraction component of the module is now capable of detecting important states associated with incidents. Work in linguistic

analysis has focused on UD-based corpora and models that have been applied to both analysis and generation. They are evaluated and reported in D5.3.

5. Crisis Classification: In this deliverable, a final version of the Crisis Classification system is presented. Generally, it encompasses technologies to strengthen the preparedness of the authorities to manage a forthcoming natural crisis event by providing early warnings to them. Furthermore, it can monitor and assess the crisis levels of the ongoing crisis event by processing dynamically heterogeneous data and providing useful information to crisis managers and decision-makers. The risk assessment process is based on a rule-based approach and was specialised depending on the specific user requirements in each of the use cases. The Crisis Classification system interacts easily with heterogeneous data sources obtaining data from sensors, outcomes of prediction and multimedia analysis in order to further analyse them, with the goal to provide assessments of the severity/risk level of a crisis. Finally, the framework is generic, which enables it to deal with multi-hazardous events.

# 9 References

Atrey, P. K., M. Anwar, H., El Saddik, A., & Kankanhalli, M. S. (2010, Nov 01). Multimodal fusion for multimedia analysis: a survey. *Multimedia Systems", 16*(6), 345--379. doi:10.1007/s00530-010-0182-0

Daiber, J., Jakob, M., Hokamp, C., & Mendes, P. N. (2013). Improving efficiency and accuracy in multilingual entity extraction. *I-SEMANTICS*, (pp. 121-124).

Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., . . . Ferrari, V. (2018). The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale.

See, A.; Liu, P. J.; and Manning, C. D. 2017. Get to the point: Summarization with pointer-generator networks. In Proceedings of the ACL, 1073–1083.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In Proceedings of the ACL, 1003–1011.

Cordeiro, S.; Ramisch, C.; and Villavicencio, A. 2016. Semeval-2016 task 10: rule-based mwe identification and predominant-supersense tagging. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), 910–917.

Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; and Dyer, C. 2016. Neural architectures for named entity recognition. In Proceedings of the NAACL-HLT, 260–270.

Honnibal, M., and Montani, I. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Daiber, J.; Jakob, M.; Hokamp, C.; and Mendes, P. N. 2013. Improving efficiency and accuracy in multilingual entity extraction. In Proceedings of the 9th International Conference on Semantic Systems (I-Semantics).

Luong, T.; Pham, H.; and Manning, C. D. 2015. Effective approaches to attention-based neural machine translation. In Proceedings of the EMNLP, 1412–1421.

Gu, J.; Lu, Z.; Li, H.; and Li, V. O. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In Proceedings of the ACL, 1631–1640.

Klein, G.; Kim, Y.; Deng, Y.; Nguyen, V.; Senellart, J.; and Rush, A. 2018. Opennmt: Neural machine translation toolkit. In Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers), volume 1, 177–184.

Schenkel, R.; Suchanek, F.; and Kasneci, G. 2007. Yawn: A semantically annotated wikipedia xml corpus. Datenbanksysteme in Business, Technologie und Web, –12.