D8.2 Training manual Volume I

Grant Agreement:	833635
Project Acronym:	ROXANNE
Project Title:	Real time network, text, and speaker analytics for combating organised crime
Call ID: **Call name:**	H2020-SU-SEC-2018-2019-2020, Technologies to enhance the fight against crime and terrorism
Revision:	V1.0
Date:	08 July 2020
Due date:	01 July 2020
Deliverable lead:	ADITESS
Work package:	WP8
Type of action:	RIA

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635.

Disclaimer

The information, documentation and figures available in this deliverable are written by the "ROXANNE - " Real time network, text, and speaker analytics for combating organised crime" project's consortium under EC grant agreement 833635 and do not necessarily reflect the views of the European Commission.

The European Commission is not liable for any use that may be made of the information contained herein.

Copyright notice

Project co-funded by the European Commission within the H2020 Programme (2014-2020)		
Nature of deliverable:	R	
Dissemination Level		
PU	Public	☒
CO	Confidential, only for members of the consortium (including the Commission Services)	□
EU-RES	Classified Information: RESTREINT UE (Commission Decision 2015/444/EC)	□
* R: Document, report (excluding the periodic and final reports) DEM: Demonstrator, pilot, prototype, plan designs DEC: Websites, patents filing, press & media actions, videos, etc. OTHER: Software, technical diagram, etc.		

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

Revision history

Revision	Edition date	Author	Modified Sections / Pages	Comments
V0.1	20 May 2020	Agelos Deligiannis (ADITESS)	All	Original draft
V0.2	19 July 2020	Anastasios Oikonomidis (ADITESS)	Chapter 3	
V0.3	23 July 2020	Erinc Dikici (SAIL)	Section 4.1	Added info on SAIL's ASR component
V0.4	24 June 2020	Tuan-Anh Hoang (LUH)	Section 4.3	Added info on LUH's network analysis components
V0.5	26 June 2020	Costas Kalogiros (AEGIS)	Section 4.5	Added info about AEGIS' FVT acting as the user interface to ROXANNE platform
V0.6	29 June 2020	PHO, IDIAP, USAAR, BUT, AIRBUS	Section 4	Input for Section 4
V0.7	29 June 2020	Theoni Spathi (KEMEA)	Section 2	Sent to ADITESS (Training literature etc.)
V0.8	07 July 2020	Theoni Spathi (KEMEA)	All	Review and final remarks
V0.9	08 July 2020	ADITESS	All	Ready for submission version
V1.0	09 July 2020	IDIAP (Petr Motlicek)	All	Proof-reading

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

Executive summary

This deliverable D8.2, is part of WP8 Field Tests, user training and continuous testing and the purpose is to document the results of the T8.2 regarding the training material of the developed tools of the ROXANNE project. This is the first volume out of the three foreseen in the project and it contains the training material that will be used particularly for the first field test. The purpose is to train the LEAs on how to use and proceed to the basic configuration of a variety of ROXANNE tools.

The training material will be revised in the second and third volume of the deliverables under the T8.2.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

Table of contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

Table of Figures

Table of Tables

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

1. Introduction

1.1. Background

Nowadays, the use of Web based Training (WbT) in distance learning education/training is considered to be an innovative method of learning. It provides new opportunities for teaching and learning and a sufficient variety of digital based means for both trainers and trainees. In WbT, the instruction could be either “synchronous”, meaning that the communication between teacher and learner is simultaneous, or “asynchronous” which means that the student is able to interact at any time, without the teacher's presence. A combination of “synchronous” and “asynchronous” modes can also be adopted for WbT instruction[1].

In the security domain, where the usage of innovative and state of the art tools becomes a necessity, the continuous training of end-users is a challenge. To this end, the application of WbT approaches can provide the necessary information of the efficient usage of new and emerging technologies. The ROXANNE project presents a great potential for WbT where the training of tools by the technology providers should take place during and after the implementation of the project.

In particular, the application of a WbT Model is suggested and will be applied in the ROXANNE project. The proposed model concerns the teaching of theoretical and technological cognitive objects like speech and natural language processing, video and geographical meta-data processing, network analysis, etc. The teaching material includes not only theory but also concretization skills that require the use of all senses, and aren’t only servile work. In addition, the process of training in such a cognitive object cannot be characterized by simple activities as memorization, rationalization and rethinking. It should also include more composite processes, such as creation, experimentation and feedback.

1.2. Deliverable Purpose and Structure

The main purpose of this document is to provide the guidelines on how to use the developed technologies during the first field trial. The high-level training material aims to train non-technology users to run the tools while the training will be given through the e-learning training platform. Additionally, this deliverable presents the training framework as well as the training procedure and evaluation.

Following a brief introduction, this document comprises of 3 sections. Section 2 provides technical details about the Training Framework which will be used for delivering the training material to end-users through remote (or physical) training sessions. The Section 3 presents the theoretical approach of training as well as the evaluation methodology. Finally, Section 4 presents the user manuals for each technology within the requirements of the first field test.

[1] Papachristos D, Alafodimos N, Arvanitis K, Vassilakis K, Kalogiannakis M, Kikilias P, Zafeiri E. An Educational Model for Asynchronous e-Learning. A case study in Higher Technology Education. International Journal of Advanced Corporate Learning 2010;3(1): 32-36.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

2. Learning Theory and Methodology

Usually the word 'learning' is automatically connected with education, school, children, students, classrooms, and teachers. However, there is an aspect that can be easily overlooked, yet based on the statistics provided by the Adult Education Survey (AES) on lifelong learning[2], almost 45% of people in the EU aged 25-64 took part in education and training, with the majority to belong to younger persons aged 25-34[3] . The concept of Adult Learning refers to the "participation of adults aged 25-64 in education and training to learning activities after the end of initial education and is a vital component of the EU's lifelong learning policy"[4]

According to Kolb (1984)[5], 'learning is the process whereby knowledge is created through the transformation of experience', thus in order to transform that experience to an impactful and engaging process, it requires the ability to have a specific plan and follow an effective methodology, enriched with all the relevant tools and individuals who will support this process. As depicted in Figure 1, the main steps to design an effective learning program can be summarized as: (a) WHO: Get to know your audience, (b) WHAT: Get to know the theoretical foundation that surround this type of audience, (c) WHY: Set the goals and objectives of your learning course (d) HOW: Select the methodology and techniques for the needs your learning course, (e) HOW MUCH: Set the evaluation criteria and feedback to the objectives and the strategy.

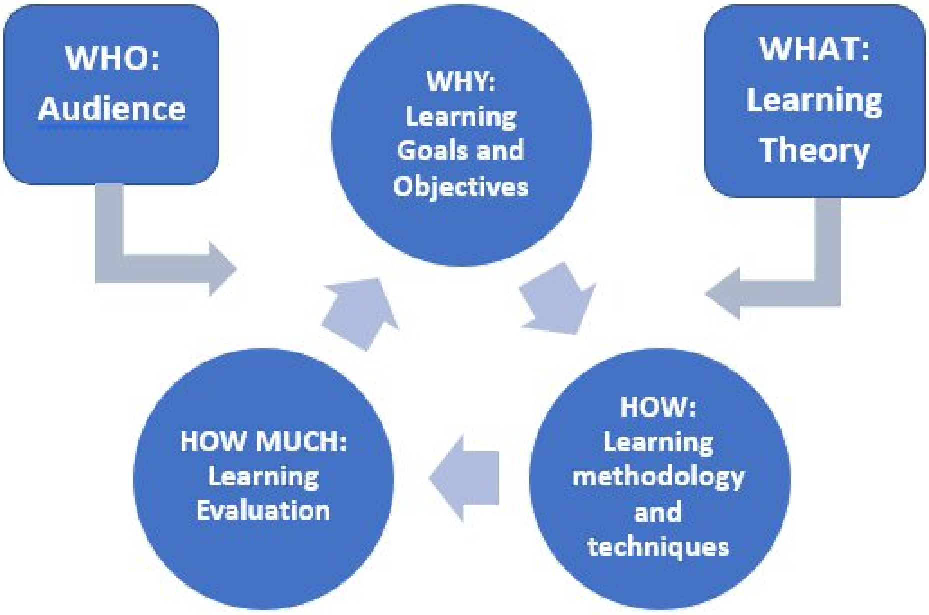

WHO: Audience
WHY: Learning Goals and Objectives
WHAT: Learning Theory
HOW MUCH: Learning Evaluation
HOW: Learning methodology and techniques

Figure 1 Steps towards an effective learning program

[2] Lifelong learning encompasses all learning activities undertaken throughout life with the aim of improving knowledge, skills and competences, within personal, civic, social or employment-related perspectives. The intention or aim to learn is the critical point that distinguishes these activities from non-learning activities, such as cultural or sporting activities. (https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Lifelong_learning)

[3] https://ec.europa.eu/eurostat/statistics-explained/index.php/Adult_learning_statistics

[4] https://ec.europa.eu/eurostat/statistics-explained/index.php/Adult_learning_statistics

[5] Kolb, D. A., 1984. Experiential learning: Experience as the source of learning and development. New Jersey: Prentice-Hall.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

2.1 WHO: Get to know your audience

To begin with, the first step for designing a specific learning course is to familiarize with your audience and its unique characteristics. For ROXANNE project, the audience are primarily adult learners stemming mostly from Law Enforcement Agencies. The term 'adult' encompasses both characteristics of age, based of course on the specific age criteria each society has set, along with other social (roles one is expected to undertake in a society like those of a parent, an employee, etc.) and psychological determents (e.g. people who can manage conflicts of intimacy vs isolation, generativity vs. stagnation, ego integrity vs. despair (based on the Erickson eight psychological stages of development[6]). The main characteristics of an adult learner, coming also from Law Enforcement Agencies, can be summarized as[7]:

- They have established **clear goals** for participating to the specific learning process, setting also certain expectations of the learning process. Personal and professional development, fulfilment of specific needs, prestige acquisition is but some of the aforementioned incentives
- They have a wide range of **diverse experiences** deriving from a range of life situations, family, relationships, business etc., which bring them to their learning experience
- They have developed their **personal preferred learning style** depending on their personal characteristics, abilities, and experiences
- They tend to present a **more active participation** in the learning activities, demanding to be treated with maturity, thus also challenging the educational content and the methodology used.
- They face **learning obstacles** deriving from social obligations, duties, or other internal psychological factors.
- They may develop **defence mechanisms and resignation**, leading them to resist sharing new insights and redefine previous knowledge, values, and habits.

As a matter of fact, based on the end-users training questionnaire, from an initial sample of 31 participants, who will be actively involved to the ROXANNE training procedures, the majority of the participants are working with the police (29%) mostly in a research position (22%) followed by management (19%), investigator (16%), forensic specialist and analyst (16%). 90% of them have not received any training at all around one or more of the named ROXANNE learning topics, while their level of knowledge varies mostly from no knowledge at all to basic knowledge[8].

2.2 WHAT: Learning Theory (Adult)

Adult learning theories provide insight to the trainers on the way adults learn, so as for the modules to be as effective and practical as possible. There have been formulated several learning theories across centuries, each one focusing on specific psychological and sociological traits of the target group they refer to. One of the most well-known theories applied to several LEA training modules is the theory of Andragogy, which establishes a (self) learning approach rather than a pure teaching approach. First proposed by Knowles in 1968[9], it focuses on the different needs of the adult learners providing them the possibility to participate in an interactive process through self-directed group discussions and active debate within the context of the classroom which leads to the continuous transformation of their experiences, thing very important for LEAs participants. According to Merriam (2001)[10] 'the five assumptions underlying andragogy describe the adult learner as someone who (1) has an independent self-concept and who can direct his or her own learning,

[6] https://www.simplypsychology.org/Erik-Erikson.html

[7] Kokkos, A. , 2005. Adult Education. Detecting the field. Athens: Metaichmio

[8] Deliverable D2.2. End User Training Requirements

[9] http://www.cfisd-technologyservices.net/uploads/5/1/5/7/51575175/l_v_cetl_facility_the_adult_learning_theory_andragogy_knowles.pdf

[10] Merriam, S. B., 2001. Andragogy and self-directed learning: pillars of adult learning Theory, New Directions for Adult and Continuing Education, 89 (1), 3-13.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

(2) has accumulated a reservoir of life experiences that is a rich resource for learning, (3) has learning needs closely related to changing social roles, (4) is problem-centred and interested in immediate application of knowledge, and (5) is motivated to learn by internal rather than external factors'[11]. The trainer's role is to facilitate learners' movement toward more self-directed and responsible learning as well as to foster their internal motivation to learn rather than act just as a lecturer or grader.

ROXANNE training process is going to adopt this training format, enhancing it with the main premises of Kolb's theory of learning named as 'Experiential Learning'. Based on the famous phrase from the Chinese philosopher Confucius (450 B.C.)[12] "Tell me, and I will forget. Show me, and I may remember. Involve me, and I will understand." this learning theory first coined in the 1970s, also adopts a hands-on approach that puts the learner at the center of the learning experience, combining it with a reflective learning style. It is represented by a four-stage learning style that includes the following steps[13] (Figure 2)

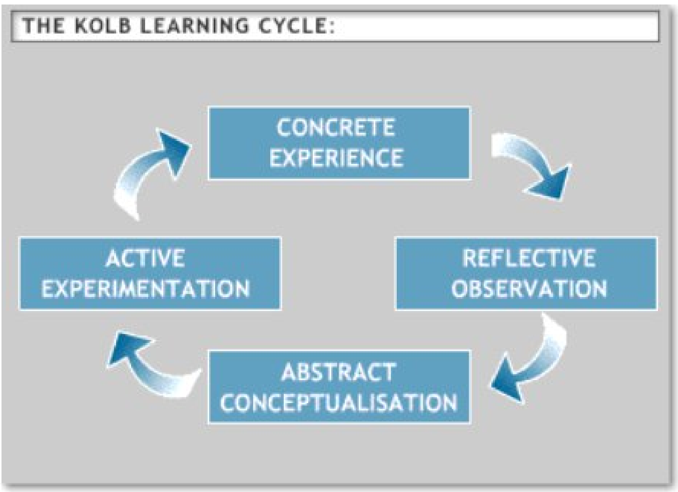


Figure 2 Kolb's Learning Cycle[14]

- **Concrete Experience** – The cycle begins with the task each individual is assigned to do/learn. Kolb underlines the fact that participants should be active learners and not passive bystanders
- **Reflective Observation** – The second step of the cycle is related to reflecting on what has been done or experienced. It is thus critical to not only create opportunities for experience-based learning but also provide time and space to encourage reflection
- **Abstract Conceptualization** – The third step underlines the process of understanding of what has happened interpreting the events and the relationships between them.
- **Active Experimentation** – The final step is putting theory into practice and proceed to the final evaluation of the learning process.

2.3 WHY: Learning goals and objectives

The cornerstone for an effective learning process is to set clear and relevant goals and objectives, which stem both from the participants and the trainers/instructors. To begin with, training goals are considered to

[11] https://elearninginfographics.com/adult-learning-theory-andragogy-infographic/

[12] Morgan. K., 2008. Experiential Perspectives. In J.M. Spector, et.al., 2008. Handbook of Research on Educational Communications and Technology, 3rd ed. London: Taylor Francis Group. P. 35

[13] McLeod, S. A., 2017. Kolb - learning styles and experiential learning cycle. Simply Psychology. https://www.simplypsychology.org/learning-kolb.html

[14] https://www2.le.ac.uk/departments/doctoralcollege/training/eresources/teaching/theories/kolb

be a broader concept, like general statements of what one hopes to accomplish as a result of training, while training objectives concern the specific results of each training module. In order to set clear objectives, one must know the needs of the project/program and the real potential of the learners. Learning objectives have to (a) clearly describe the skill or behaviour that has to be achieved after the end of each training module, (b) indicate the conditions where the trainees will need to demonstrate their knowledge and skills acquired and (c) be measurable in terms of final performance[15].

ROXANNE training modules will take place before each of field test, also having the potential to be repeated during the three years of continuous testing of the ROXANNE platform. Table 1 summarizes the objectives for each iteration focusing on the first field tests.

Table 1 ROXANNE 1st Field Test - Training Modules

Module	Objective
Audio Pre-Processing	At this phase of the project, the purpose of the training module is the familiarization of LEAs with the proposed by ROXANNE technologies. The basic configuration of the tools as well as the set of input files is within the scope of training process.
Speaker, Age and Gender Identification	
Language Identification	
Automatic Speech Recognition	
Entity Detection	
Topic Detection	
Network Analysis	
Input/output GUI	

2.4 HOW: Learning Methodology and Techniques

As Rogers and Horrocks (2010)[16] discuss three adult education sectors can be distinguished (Figure 1), the formal which consists of courses and classes run by schools, colleges and other agencies that belong to the educational system, the extra formal (non-formal) that consists of classes and courses provided from agencies outside the educational system (e.g. training agencies, government departments etc.) and the informal that refers to educational activities engaged by voluntary agencies and informal groups.

[15] https://eclearn.emmanuel.edu/courses/1285497/pages/how-to-write-measurable-learning-objectives

[16] Rogers, A., & Horrocks, N., 2010. Teaching adults. McGraw-Hill International.

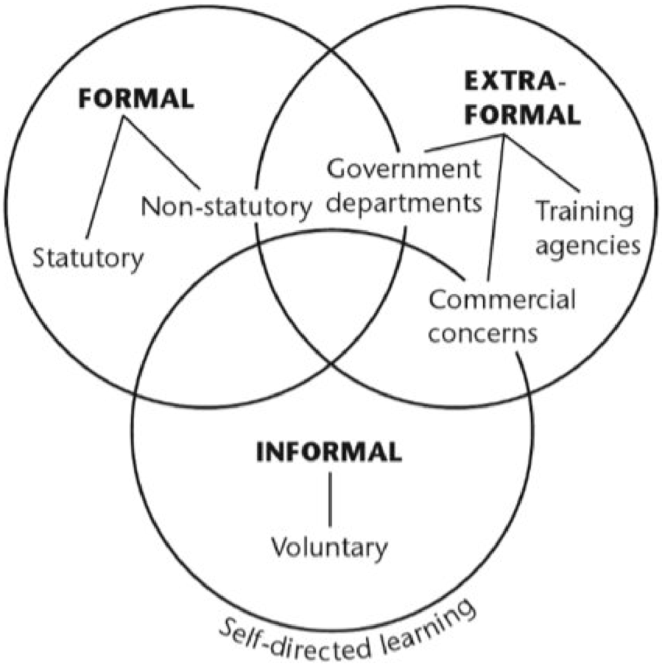


Figure 3 Sectors of Adult Education

ROXANNE learning sector lies primarily to the second group. Selecting specific learning methodology and techniques is the concrete outcome of the learning objectives and final outcomes, the content, the characteristics of the participants the time constraints as well as the potential of implementation and available infrastructure.

According the final outcomes of the deliverable D2.2. *End user Training requirements*, the 31 participants have reported that training for the ROXANNE platform must include e-learning and blended learning (e-L and classroom) potential, it must be adaptive and interactive to the trainee and its needs, depicting the full potential of the system and allowing them to work on realistic scenarios in order to better demonstrate its impact. Other educational features that LEAs consider useful for an effective training procedure are incremental webinars that are adaptive to their own time and pace. Furthermore, training content should be carefully designed for participants with special educational needs, should be multilingual, provide the theoretical background and allow participants to learn the system's components individually.

2.5 HOW MUCH: Learning Evaluation

The whole process would not be complete if it did not entail the stage of assessment and evaluation. This specific step has also been assessed as a Must-Have by the 31 respondents of the end user training requirements questionnaire (D2.2 *End user Training requirements*). This stage has as its main objective through the assessment of the participants' performance to validly and reliability estimate the degree of knowledge, skills and competences acquired through participation in the training activity, including also the level of performance both from the trainer and from the techniques and methods used during each course. The final outcomes will feed back to the process to re-evaluate the learning goals and objectives as well as the methodologies and tools selected for the exact participating audience. Assessment can be both formative (during the learning process) and summative (after the learning process)[17]. Common evaluation

[17] Mavrogiorgos, G., 2003. Assessment methods and the certification process of the candidates. Ministry of Labour and Social Insurance.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

tools can include questionnaires, comprehension checks, in-class activities and deliverables, quizzes, online assessments, and reflective diaries as tools for self-assessment on behalf of the participants.

For the ROXANNE project an online education platform will be used, where trainees will be able to interact with trainers, view related material of every topic individually and go through online workshops. The same platform allows the collection of feedback regarding performance of every trainee and offers to the trainers an easy and reliable way to evaluate everyone at the end of the training period.

3. Training Framework

For the purposes of training the features and use of ROXANNE system to several trainees coming from different backgrounds and LEAs, an online educational platform will be used. As the levels of experience in various ROXANNE principles and characteristics were determined through the relevant questionnaire described in D2.2 *End user Training requirements*, a well-known for its effectiveness platform should be used, flexible enough to allow the design and implementation of courses who will serve well both instructors and learners. Such a platform is Moodle (Figure 4), which is widely used by several universities and other educational organizations worldwide.

The educational process will be formed in training modules. Educators and trainees will be assigned to each one of them and receive the relevant roles. Each module will operate as a micro Web site within the platform where all participants will be able to communicate with each other through the module's blog, which will be visible only to them. Educators may post questions for evaluation and practicing purposes. All material will be available according to educators' decision through the module for easy downloading, while time restrictions may apply for all trainees regarding availability.

External resources (Web links, videos, pictures etc.) are possible to get embedded in the material for security reasons or point trainees directly to them.

Workshop in different forms can be offered for practicing; from simple questions and answers, to more complex quizzes where each trainee will be able to test their knowledge on the module content upon finishing their study of it. In case of homework, where this may apply, a prescheduled time for submission and uploading widget, allows educators to assign tasks for the trainees and collect feedback in an easy to use fashion.

Trainees will be grouped in cohorts according to their organization, or any other criteria. Through this way, educators will have the possibility to assign different tasks to different groups.

If necessary for the benefit of educational process, educators will assign to each trainee a total scoring reflecting their performance at the end of the teaching period, or individually for every separate task and/or homework. Moodle's scoring features offer a reliable process of scoring and pass grades to each member, keeping it hidden from the other members or not, according to the instructors or group's decision.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

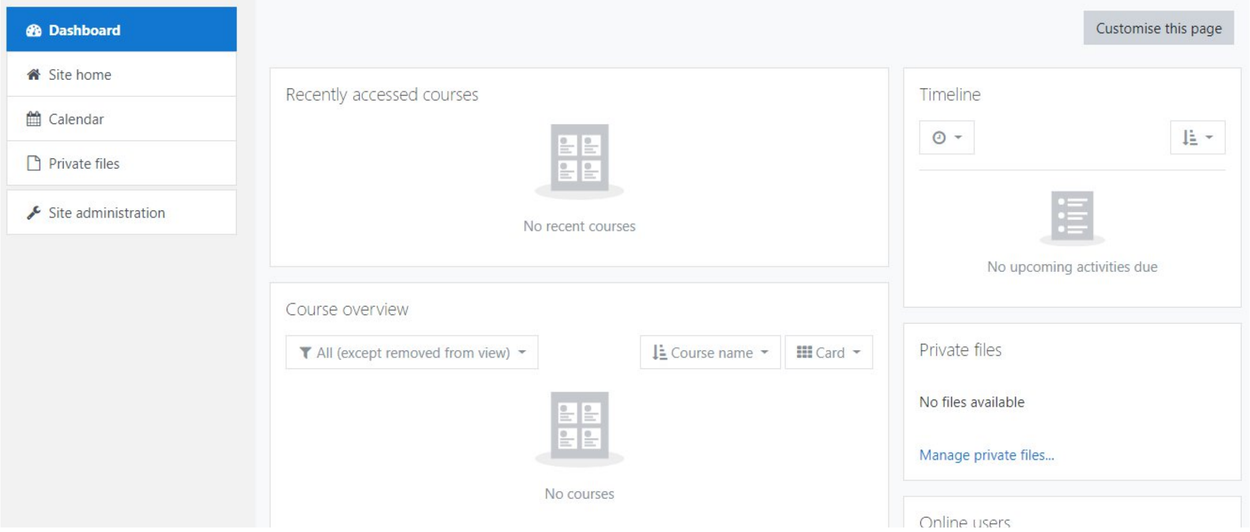


Figure 4 Web Interface of Training Platform

4. Training Modules

4.1 Audio Pre-Processing

The Audio Pre-Processing module is responsible for ingesting the input audio files, estimation of their quality (in order to decide whether or not to process it), determination of voice activity segments (i.e.. parts in the file where speech is present) and eventually for diarization - for mono-recordings including the voices of multiple speakers, diarization determines which speaker speaks when.

4.1.1 Data Input

The input to audio processing modules is audio. Supported audio format are:

- WAVE (*.wav) container including any of:

  unsigned 8-bit PCM (u8)

  unsigned 16-bit PCM (u16le)

  IEEE float 32-bit (f32le)

  A-law (alaw)

  μ-law (mulaw)

  ADPCM
- FLAC codec inside FLAC (*.flac) container
- OPUS codec inside OGG (*. opus) container

Other audio formats must be converted using external tools. Speech Engine (SPE) server can be configured to support automated conversion on background, see SPE configuration hints.

Tools for converting other than supported formats to supported are ffmpeg (http://www.ffmpeg.org) or SoX (http://sox.sourceforge.net/). Both are multiplatform software tools for MS Windows, Linux and Apple OS X.

Example of usage:

ffmpeg

```
ffmpeg -i <source_audio_file_name> <output_audio_base_name>.wav
```

It causes than any supported format/codec audio file will be converted to normalised WAV audio format in 16-bit PCM little endian as it is default system. For more parameters please check manual pages.

SoX

```
sox <source_audio_file_name> -b 16 <output_audio_base_name>.wav
```

Number of bits defined by -b parameter must be specified.

It is expected that media files can come with meta-data files like csv. Meta data might be recording date and time, geo-location, IMEI of device etc. Metadata are not directly used in speech technologies, but can serve as inputs in further steps in processing chain.

4.1.2 Output

Speech quality estimation

SQE outputs a range of characteristics on the input signal and an overall score (Table 2):

Table 2 Audio Pre-Processing Input Parameters

waveform_clipped_length	length of the clipped signal (amplitude is greater than specific maximum value) in seconds
waveform_clipping_threshold	threshold for measuring the clipped signal length
waveform_kurtosis	"peakedness" of the probability distribution of signal samples
waveform_length	total length of the signal in seconds
waveform_max_abs_value	maximum amplitude
waveform_max_value	maximum sample value
waveform_mean	mean value of samples
waveform_min_abs_value	minimum amplitude
waveform_min_value	minimum sample value
waveform_n_bits	number of bits used to encode the waveform
waveform_n_levels	number of signal levels (different values of signal)
waveform_sample_freq	sampling frequency
waveform_skewness	asymmetry of the probability distribution of signal samples
waveform_snr	signal-to-noise ratio in dB based on gamma vs. gaussian distribution comparison
waveform_standard_deviation	standard-deviation of samples
wfilter_filtered_length	total filtered length (silence + intermittent + technical) in seconds
wfilter_filtered_ratio	ratio of filtered signal length and total length

wfilter_intermittent_noise_length	length of intermittent noise in seconds
wfilter_intermittent_noise_ratio	ratio of intermittent signal length and total length
wfilter_speech_signal_length	total unfiltered length in seconds (waveform_length - wfilter_filtered_length)
wfilter_silence_length	length of silence (based on energy threshold) in seconds
wfilter_silence_ratio	ratio of silence length and total length
wfilter_snr	signal-to-noise ratio in dB based on energy threshold (not accurate, use waveform_snr)
wfilter_technical_signal_length	length of technical signals (tones, wide-band noise, ...) in seconds
wfilter_technical_signal_ratio	ratio of technical signal length and total length

The global score (range <0;100>) is specified at the last line of the output file. Minimum variable score is taken as a global score:

Global score = MIN (variable_score)

The global score is based on waveform_n_bits and waveform_snrvariables.

It is recommended to postprocess audio files based on the global score thresholds; the ROXANNE platform is configured based on Table 3.

Table 3 Audio Pre-Processing Global Score

Global Score	Conclusion
Less than 50 %	quality is poor for speech technology processing
Between 50% and 75%	quality is sufficient for speech technology processing
Greater than 75%	quality is good for speech technology processing

Voice Activity detection

VAD has a text or JSON format determining the start and end of voice, resp, silence region, it can be visualized in the following way:

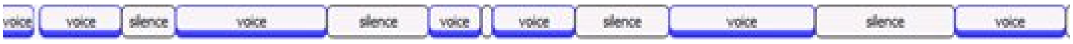


Figure 5 Voice Activity Detection Visualization

Speaker diarization

Similarly to VAD, the output of speaker diarization has a text or JSON format determining starts and ends of segments, and speaker identity. Note that in this stage, the system has no way to determine a true speaker identity (this is a task for a speaker identification module), so that the output uses symbols such as ‘1’, ‘2’ etc for individual speakers:

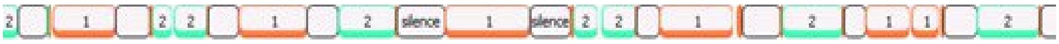


Figure 6 Speaker Diarization Visualization

4.1.3 Configuration Parameters

Note that all audio pre-processing modules have very similar parameters and only examples are given here. The actual configuration parameters can change in different versions of ROXANNE platform.

Common parameters:

```
  -i, -in-file file           input file
  -l, -in-list file           list of input files
  -d, -in-dir dir             input directory
  -e, -in-ext str [raw,wav]   extensions of input files (comma separated)
  -f, -fmt fmt [lin16]        waveform format (lin16, lin8, lin8offset, alaw, mulaw)
  -n, -nchannels num [1]      number of channels in audio files
  -s, -sample-freq num [8000] sampling frequency of audio files
  -no-in-errs                 suppress input file errors
```

Speech quality estimation

No special configuration parameters

Voice activity detection

No special configuration parameters

Speaker diarization

```
 -total-speakers num           total number of speakers
 -max-speakers num    [2]       maximal number of speakers
 -max-avg-dist num    [0.75]    maximal average distance between speakers
```

4.2 Speaker, age and gender identification

Speaker identification (Voice Biometric technology) can be used to quickly search for and accurately identify speakers within large amounts of audio recordings. The basics of speaker identification is the derivation of voice-prints - compact and fixed-size representation of recordings, that can be then used for very fast speaker comparison. The voice-print contains also information about the gender and age of the speaker, therefore, age estimation and gender recognition use the same set of voice-prints. Phonexia SID version 4 (SID4) is used for all operations with voice-prints, their actual comparison, age estimation and gender detection.

4.2.1 Data Input

Voice-prints are extracted from pre-processed (Section 4.1) audio files using voice-print extractor. They are binary files containing the speaker information (useful for speaker comparison, age estimation and

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

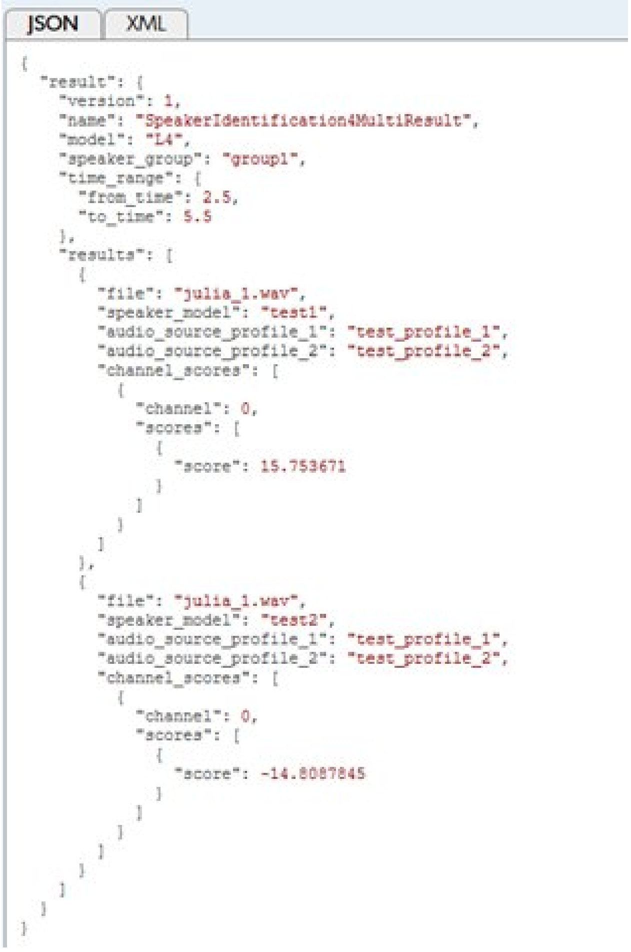
gender detection) complemented with meta-information. The tool **vpextract4** is used to extract voiceprints from speech recordings. Tool **vpinfo** can be used for listing information about voiceprints.

4.2.2 Output

Speaker identification

The crucial output of the engine is the score comparing the speaker in two audio files or audio segments. In ROXANNE platform, we are using a matrix of such scores to infer the criminal network. Figure 7 presents a JSON example of comparison of two files. Such files contain the raw (log-likelihood) score that is not normalized to 0...100% range.

The “so-called verification score” can be easily translated into an identification score (e.g. rank), allowing to identify targeted speaker in the whole database (i.e. across all enrolled speakers).



```
{
  "result": {
    "version": 1,
    "name": "SpeakerIdentification4MultiResult",
    "model": "L4",
    "speaker_group": "group1",
    "time_range": {
      "from_time": 2.5,
      "to_time": 5.5
    },
    "results": [
      {
        "file": "julia_1.wav",
        "speaker_model": "test1",
        "audio_source_profile_1": "test_profile_1",
        "audio_source_profile_2": "test_profile_2",
        "channel_scores": [
          {
            "channel": 0,
            "scores": [
              {
                "score": 15.753671
              }
            ]
          }
        ]
      },
      {
        "file": "julia_1.wav",
        "speaker_model": "test2",
        "audio_source_profile_1": "test_profile_1",
        "audio_source_profile_2": "test_profile_2",
        "channel_scores": [
          {
            "channel": 0,
            "scores": [
              {
                "score": -14.8087845
              }
            ]
          }
        ]
      }
    ]
  }
}
```

Figure 7 Speaker Identification JSON Example

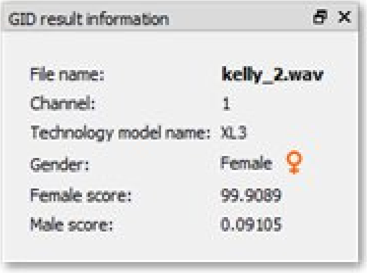
Age identification

The output contains a list of input segments each with the estimated age.

Gender identification

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

The output contains a list of input segments each with the estimated gender and score, see example in figure:



File name: kelly_2.wav
Channel: 1
Technology model name: XL3
Gender: Female ♀
Female score: 99.9089
Male score: 0.09105

Figure 8 Gender Identification Visualization Example

4.2.3 Configuration Parameters

The configuration parameters for the Speaker, Age and Gender identification tool can be found below.

Voice-print extraction

```
output setting:
   -o, -out-file file            output file
   -D, -out-dir dir              output directory
   -E, -out-ext str [vp]         extension of voice-print files

 diarization options:
  -total-speakers num           total number of speakers
  -max-speakers num    [6]      maximal number of speakers
   -max-avg-dist num    [0.75]  maximal average distance between speakers

 calibset options:
  -calibset-total-chunks num [6]  total chunks in calibset voiceprint
```

Speaker recognition - voice-print comparison

```
  input:
     -i, -in-file                compare two voice-print files
     -l, -in-list                compare two lists of voice-print files
     -d, -in-dir                 compare two directories of voice-print files
     -e, -in-ext str [vp]       extension of voice-print file

   output:
     -o, -out-file file          output score file
    -suppress-too-short        suppress the warnings about short voiceprint and
                                 '-inf' values in output for cases, where short
                                 voiceprints are compared;
                                 results may be less reliable for such voiceprints

    -get-file-names             dump voice-print file names to the score file
    -get-file-paths             dump voice-print file names with full path to the score file
    -get-n-best num             for each voice-print from input1 dump only N
                                 best scores from input2
    -get-top-scores num         for each voice-print from input1 dump only scores
                                 from input2 which are greater than threshold
    -score-sharpness num [1.0] score sharpness (positive number)
    -score-scale num [1.0]     score scale (applied only when voiceprint does not
                                   contain calibration data; overrides system/cmp_score_scale
                                   value from configuration file; positive; non-zero)
    -score-shift num [0.0]     score shift (applied only when voiceprint does not
                                   contain calibration data; overrides system/cmp_score_shift
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

```
                                   value from configuration file)

                                 NOTE: Score shift and scale are applied according
                                 following formula:
                                   score = (original_score * scale) + shift

     -F, -out-fmt columns       enable column output format columns to print are
                                 specified by string of the characters below, e.g. lsn)
        s                        raw score
        n                        score normalized to <0, 100>
        k                        speech length of input1 voice-print
        l                        speech length of input2 voice-print

 training mode (enabled by option -M):
    -adapt-by-counts            use counts (of voice-prints) for adaptation
    -adapt-const num [1.0]     adaptation constant <0.0, 1.0>
                                 0.0 ... old models are used only
                                 1.0 ... new models are used only
   input:
     -l, -in-list               train from two-column list file; speaker name in
                                first column, voice-print file in second column
     -d, -in-dir                train from directory with subdirectories each
                                 corresponding to a single speaker
     -e, -in-ext str [vp]       extension of voice-print file

   output:
     -M, -out-model-dir dir     output model directory
```

Age identification

```
 output:
   -o, -out-file file             output score file
   -F, -out-fmt columns           enable column output format, columns to print are
                                   specified by string of the characters below, e.g. alc
     a                            age
     l                            speech length
     r                            record length
     c                            channel number
                                   (if -per-channels option is enabled)

   -suppress-too-short            suppress the 'too short' output
```

Gender identification

```
output:
   -o, -out-file file    [stdout]  output score file
  -get-both-scores                produce scores for both genders
  -score-sharpness num            score sharpness (positive number)
  -gender-balance num   [0.0]      gender balance (-1.0, 1.0)
                          -1.0 ...  100% males
                           0.0 ...  equally balanced
                           1.0 ...  100% females
  -unk-class-thr num    [80.0]     unknown class (U) score thr. (50.0, 100.0)
                          50.0 ... unknown class off
                          >50.0 ... unknown class if less than thr.
```

4.3 Language identification

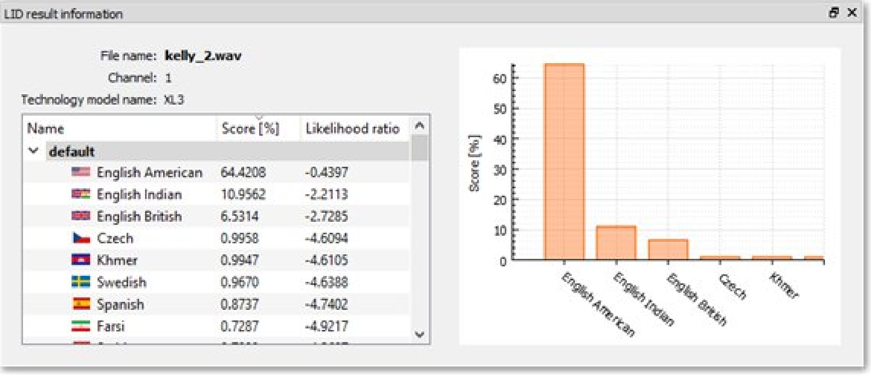
Language Identification (LID) helps users to distinguish the spoken language or dialect[18]. It will enable users' system to automatically route valuable calls to their experts in the given language or it will be sent to another modules or software capable of analysing it. LID technology is very fast. The module is delivered with 50+ pretrained language models. Users can also create a completely new language models using their audio recordings. Similarly, to speaker recognition, a language print extraction (deriving a fixed-length vector representation from a speech segment) begins the process, the language-prints are then processed with a classifier that can operate on a full or limited set of languages.

4.3.1 Data Input

Language-prints are extracted from pre-processed (see section 4.1) audio files using language-print extractor **lpextract**. They are binary files containing the language information. Tool **lpinfo** can be used for listing information about language-prints.

4.3.2 Output

The language information is available in the form of languages and associated raw and normalized scores in text, XML or JSON format. The following figure provides a visualization of LID output (the actual visualization in the ROXANNE platform may differ):



LID result information

File name: **kelly_2.wav**
Channel: 1
Technology model name: XL3

Name	Score [%]	Likelihood ratio
default		
English American	64.4208	-0.4397
English Indian	10.9562	-2.2113
English British	6.5314	-2.7285
Czech	0.9958	-4.6094
Khmer	0.9947	-4.6105
Swedish	0.9670	-4.6388
Spanish	0.8737	-4.7402
Farsi	0.7287	-4.9217

Figure 9 Language Identification Visualization

4.3.3 Configuration Parameters

The configuration parameters of the language extractor tool can be found below.

Language-print extractor

```
output:
   -o, -out-file file            output file
   -D, -out-dir dir              output directory
   -E, -out-ext str [lp]         extension of language-print files
   -archive file                 pack language-prints to an archive
   -out-calib file               make calibration file instead of language-prints
```

Language identification module

[18] The question of what is a language and what is a dialect is not solved by engineers, we simply rely on the availability of data and user configuration.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

```
output:
   -o, -out-file file          output score file
   -F, -out-fmt columns        enable column output format, columns to print are
                                specified by string of the characters below, e.g. lsn
     s                         raw score
     n                         score normalized to <0, 100>
     l                         speech length
     r                         record length
     c                         channel number
                                (if -per-channels option is enabled)

  -get-all-scores             produce scores for all languages
  -suppress-too-short         suppress the 'too short' output
   -get-n-best num             output only N best scores
  -get-top-scores num          output only scores greater than threshold

 training:
   -train                      enable training
   -M, -out-model-dir dir      output model directory
```

4.4 Automatic Speech Recognition

The aim of automatic speech recognition (ASR), also known as speech-to-text (S2T), is to create a transcription of what is being said in an audio recording.

In the first version of the ROXANNE platform to be presented in the 1st Field Test event, the ASR component is based on Media Mining Indexer, SAIL's media processing and rich transcription tool. By processing an input file with this component, the users should expect to get an automatic transcription of the spoken content, which contains the recognized words and associated statistical information (see Section 4.4.2 for details).

4.4.1 Data Input

The ASR component requires the path to the audio file and the spoken language code as their inputs. The input audio file is the outcome of the audio pre-processing component (Section 4.1), hence the path to it will be managed by the ROXANNE platform itself. The language code is a string from a standardized catalogue such as ISO 639-3, which defines the language and dialect of the speech which is being processed.

The ASR component in the first version of the ROXANNE platform supports only International English, since all datasets to be processed in the 1st Field Test are in this language. SAIL already has ASR systems available for 25 languages and these will be deployed in the ROXANNE platform, depending on the datasets and the end-users' interest, in the future.

Next versions of the ROXANNE platform will also feature the bridging between the LID (Section 4.3) and ASR components, so that the identified language is automatically transferred as input to the recognizer.

4.4.2 Output

The native output of the ASR component is a JSON formatted text file which contains the following information:

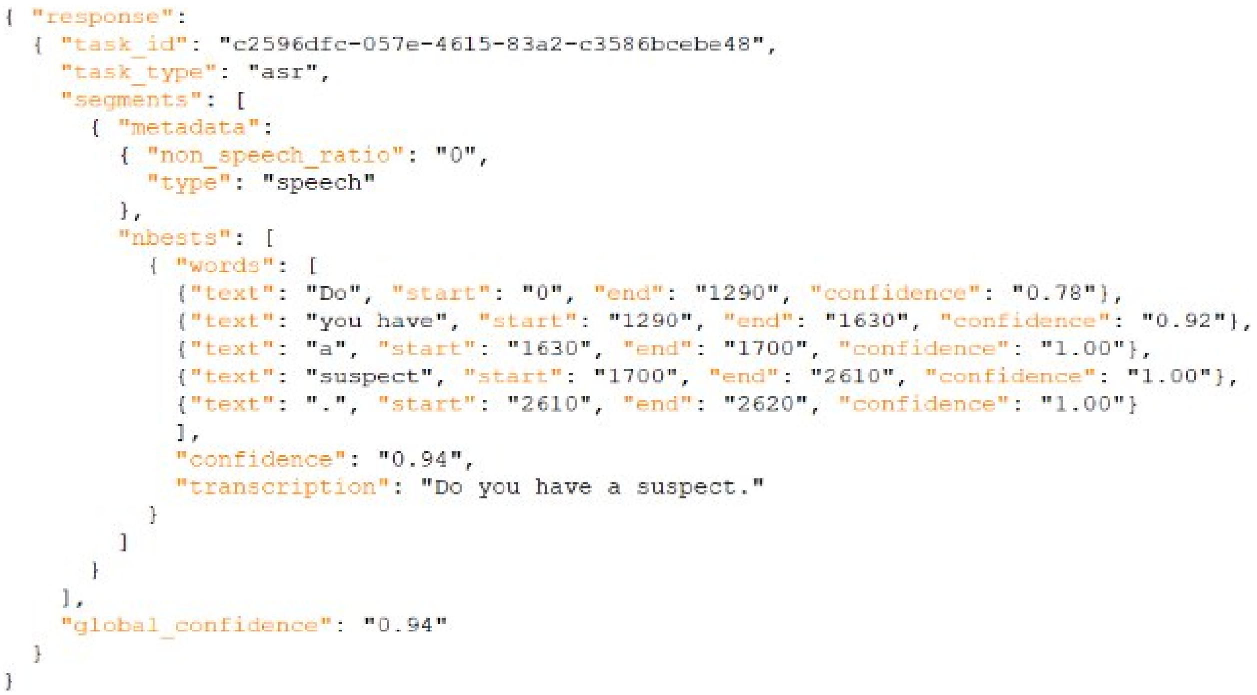
- Task identifier: A unique UTF-8 string to identify the task, for logging purposes
- Type of the task: The string "asr" to denote the type of the task
- A list of segments (if there is only one segment, it should span the whole file)

- Segment-dependent metadata, such as characteristics of audio, estimated noise level and content type

The following items pertain only to segments containing speech

- N-best hypotheses, each one containing
    - Recognized word, its start and end offsets in the source file (timestamps in milliseconds) and its confidence score in (0.00,1.00]
    - N-best score
    - Transcription: Text of the recognized speech (the first letter of each segment is capitalized)
- Global confidence score in (0.00,1.00]

A segment is a continuous stretch of audio in which the acoustic conditions is computed to be stable, and runs most likely from a point of silence (pause) to another. It may be labelled as “speech”, “music” or “noise”. Please note that a segment labelled as “speech” not necessarily corresponds to a complete sentence; it may span multiple sentences if the speaker is not giving any breaks in between, or may end in the middle of a sentence if the speaker pauses (breathes, hesitates) too much. A confidence score can be seen as a floating-point number which indicates how likely the recognizer believes that the recognized word is correct (the higher the better). The following figure shows an example output of the JSON output.



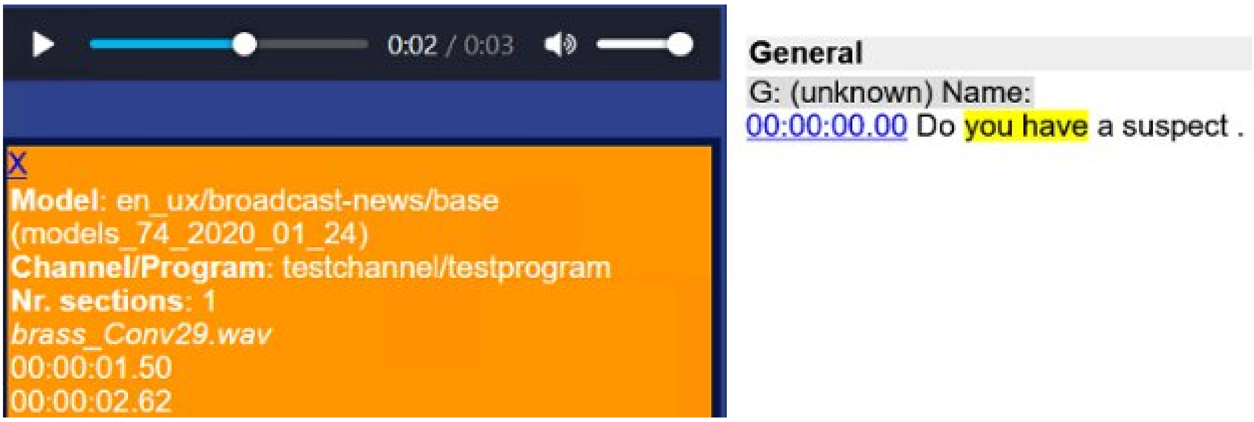
```
{ "response":
  { "task_id": "c2596dfc-057e-4615-83a2-c3586bcebe48",
    "task_type": "asr",
    "segments": [
      { "metadata":
        { "non_speech_ratio": "0",
          "type": "speech"
        },
        "nbests": [
          { "words": [
            {"text": "Do", "start": "0", "end": "1290", "confidence": "0.78"},
            {"text": "you have", "start": "1290", "end": "1630", "confidence": "0.92"},
            {"text": "a", "start": "1630", "end": "1700", "confidence": "1.00"},
            {"text": "suspect", "start": "1700", "end": "2610", "confidence": "1.00"},
            {"text": ".", "start": "2610", "end": "2620", "confidence": "1.00"}
            ],
            "confidence": "0.94",
            "transcription": "Do you have a suspect."
          }
        ]
      }
    ],
    "global_confidence": "0.94"
  }
}
```

Figure 10 Automatic Speech Recognition JSON Output Example

The inclusion of timestamps in the JSON output makes it easy to locate a specific word or phrase in the recording and listen to it in context. The following figure shows an example visualization of the ASR output (from the Media Mining Indexer), in which words are highlighted with respect to time in a browser window.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

Although the GUI of the ASR module is not available at the time of writing of this deliverable, the final ROXANNE platform may also feature a similar interface.



Model: en_ux/broadcast-news/base (models_74_2020_01_24)
Channel/Program: testchannel/testprogram
Nr. sections: 1
brass_Conv29.wav
00:00:01.50
00:00:02.62

General
G: (unknown) Name:
00:00:00.00 Do you have a suspect .

Figure 11 Automatic Speech Recognition Visualization Example

4.4.3 Configuration Parameters

The ASR component as implemented in the first version of the ROXANNE platform does not require any configuration, as all configuration and runtime parameters have already been optimized for the platform by by SAIL beforehand.

Nevertheless, it must be noted that an ASR system has a statistical nature and hence, is bound to make errors. The transcription accuracy of an ASR system is affected by many factors, including but not limited to

- acoustic conditions of recording (type and location of the microphone, background noise/music, reverberation, interference, etc.),
- speaker's speaking style (accented/dialectal speech, intonation, emotion, speed, hesitations, etc.),
- spoken content and language use (topic, use of infrequent words, specific terms, jargon, abbreviations and proper nouns, incomplete sentences)

Another point to note is that the speech recognizer will never invent new words on the fly (with the possible exception of compound words). Therefore, it will never be able to recognize previously unseen words correctly and always produce recognition errors in case such a word is uttered. Not only will the unknown word itself be mis-recognized, the mis-recognition itself will also affect the language model context and cause additional recognition errors (as a rule of thumb one unknown word causes one and a half recognition errors on average).

In the future, depending on the future use-cases of the ROXANNE platform and the end-users' interest, SAIL may in provide its Language Model Toolkit, an extension to its Media Mining Indexer, which allows the users to modify their ASR components by adding new vocabulary and building custom language models.

4.4 Entity detection

Named-Entity Recognition (NER) will highlight the named-entities in the text. Named-entities can be person names, locations, organizations etc. It can help users go through the text documents faster. For example, by glancing the highlighted entities a user can quickly decide whether the current text document is relevant for him/her. And if it is relevant, the entities can let the users to directly focus on the related parts of the document. In short, NER should significantly reduce the time for processing the text documents. Also, the detected entities can be used for further analysis, e.g. to provide features for network analysis.

4.4.1 Data Input

Currently our NER component supports only English but we will support more languages later.

The input for the NER component is the following:

- job id: A unique UTF-8 string to identify the task, for logging purposes
- input language id: it tells the component in what language the input is.
- Inference setting name: our component will support multiple NER module from powerful ones to compact ones. Also, we can support
- target entities: list containing entities the user want to detect. Default will be a list containing all entities the model can support.
- Input text: the text on which we apply NER

Our component is integrated in a docker environment and supports JSON input, by sending a JSON file containing the input our component will start processing. An example of the input JSON file is shown below:

```
[
 {
   "job_id":"ee76a60a-b32c-4151-810e-c21817e3d142 ",
   "src_language":"en ",
   "inf_setting_name":"base ",
   "Input text":"Fred and Mary got married, ..."
  }
]
```

4.4.2 Output

The native output of the NER component is a JSON formatted text file. For each detected entity, we will output it's type (person name, organization etc.), position in the input text and the confidence of our detection. We provide an example output file here:

```
[
  {
    "task_id": "30379fcd-273d-47a9-ad37-278f56a55b62",
    "task_type": "ner",
    "entities": [
      {
        "token": "Fred",
        "uuid": "40fd5f77-5564-44e1-9aee-4d0dd3723ee1",
        "type": "PERSON ",
        "confidence ": 0.9,
        "segments": [
          {
            "start": 1,
            "end": 1,
          }
        ]
      },
      {
        "label": "Mary ",
        "uuid": "40fd5f77-5564-44e1-9aee-4d0dd3723ee2",
        "type": "PERSON",
        "confidence ": 0.9,
        "segments": [
          {
            "start": 3,
            "end": 3,
          }
        ]
      }
    ]]
  }
]
```

4.4.3 Configuration Parameters

To apply our NER component, we need to input a list of parameters. We pack all of them into a single configuration file to ease the usage. An end-user only needs to specify the name of the configuration file. The parameters packed into the configuration file is:

- model name: we support multiple models for NER. Some models are more powerful than the others but require more time to process.
- word embedding path: to process the text in a mathematical manner, we convert every word in the text into a high dimensional vector. This is called the embedding of a word. We could use different embeddings for processing.
- Embedding vector size: the size of the word embedding
- number of labels: number of entities we are going to support.

4.5 Topic detection

Generally speaking, the Topic Detection task refers to automatic techniques for locating topically related material in streams of data such as news wire and broadcast news, twitter feeds, etc. Topic Detection is a Natural Language Processing (NLP) research field that aims at generating automatic tools for extracting meaning from texts by identifying recurrent themes or topics.

The Topic Detection in ROXANNE is based on unsupervised approach. The main idea is to first identify the underlying concepts contained in a given training dataset. For this, a semantic analysis (SA) process is applied for learning words representations that will later allow to generate sets of semantically associated words. Once the main concepts are and its corresponding semantically associated words are learned, it is possible to infer which are the more salient topics in the training dataset.

The Topic Detection module can help LEAs to quickly identify the main themes present in a large collection of texts, and it also helps to categorize them for more deeper analysis. The Topic Detection module follows an unsupervised approach; hence it does not require labelled data to train the model, and can perform efficiently in narrow domains with multilingual support and a flexible configurations parameter.

4.5.1 Data Input

TRAINING FILE FORMAT

The format of the input file should be two column CSV file (Comma Separated Values) type. First column is considered as the file ID, and the second column is considered as the content of the file. See the following example:

```
id1,help me oh god help me somebody please please oh god somebody help me please help me help me help me help me

id2,heads  up the press is going to be all over this one four dead mother father  two teenage boys the sisters were luckier teen girl heard a noise hid in  the closet alerted the neighbors after all the shouting was done  younger sister 's over there they could n't have killed their father tag  team soaking wet they 've been inside

id3,hey did the count change since you called me what 's the matter with your guys
```

API Details

Get Topic This API returns a single topic for the passed text content with the language. It uses the existing trained model (LDA) for inferencing a topic. If the input text contains a single quote (‘) then add (\’’) as shown below.

The possible topics for the CSI dataset are “Crime”, “Crime Scene Investigation ”, “Murder”, “Forensic Evidence”, “Detective”, “Investigation”, “Crime Lab”, “Investigator”, “Scene of the Crime”, and “Murder Investigation”.

Available APIs

- Get the training configuration parameter details.
- Set the training configuration parameters.
- Get the training model result (Training stage)
- Get Inference configuration parameters.
- Defining/Uploading the document to be classified/categorized.
- Get an Inference result.
- Get Topic
- Get Topic list

Example1

curl -X POST -H "Content-Type: application/json" -d '{ "content": "i got the blood samples on way to lab you mean the blood swirls next to father'\"s body in boys room i studied pictures of the manson murders this is n'\"t butter it'\"s imitation what'\"s your take explains why the blood is confined to bed and floor under it gave his life for the little girl there should be more blood the first suspect well she'\"d need help maybe a boyfriend", "language": "en" }' http://localhost:5000/getTopic

4.5.2 Output

Output:

```
[
  {
    "topic": "Scene Of The Crime"
  }
]
```

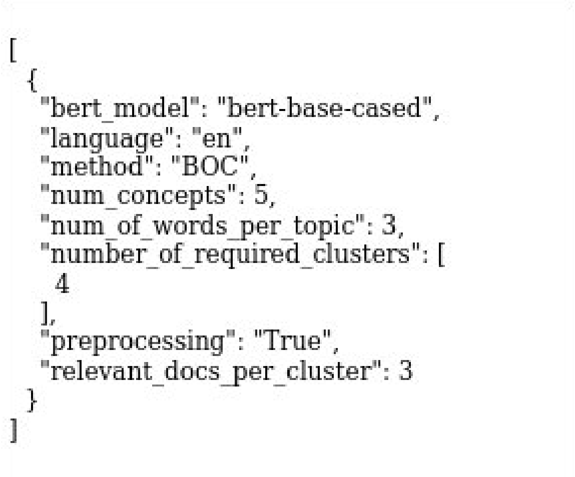
4.5.3 Configuration Parameters

Set the training configuration parameters. Except for "number_of_concepts" and "number_of_required_clusters", all other parameters are single value. Follow the below format during parameter setting.

curl -X POST -H "Content-Type: application/json" -d '{ "method": "BOC", "preprocessing": "True", "number_of_concepts" : 5, "number_of_required_clusters" : [4], "num_of_words_per_topic" : 3, "relevant_docs_per_cluster" : 3, "language":"en" }' http://localhost:5000/setTrainConfig

- **method**: Possible values for the method are "BOC", "LDA", or "LSA".
- **preprocessing**: Normally, preprocessing is required, If set to "False", means that input documents won't be cleaned. Better performance is obtained with preprocessing set to "True". Preprocessing means also preserving only NOUNS from the input text.
- **number_of_concepts:** number_of_concepts represents the dimensionality to be considered when representing documents. We demonstrate that lower values lead to better performance. If several values want to be tested, you should add possible values within double quotes separated by a comma, eg. number_of_concepts" : ["5", "10", "15"]
- **number_of_required_clusters:** number_of_required_clusters represents the number of topics (labels) that will be inferred from the data. If several values want to be tested, you should add possible values within double quotes separated by a comma, eg. number_of_required_clusters" : ["6", "8", "10"]
- **num_of_words_per_topic:** num_of_words_per_topic indicates how many words are required for describing each concept.
- **relevant_docs_per_cluster:** relevant_docs_per_cluster is used when obtaining the most salient documents for each cluster. Thus, this number indicates how many salient documents are required to retrieve. A file showing this information is saved in the /log folder.
- **language**: is a parameter used to define the language of the documents and the necessary tools for processing the information. Currently, supported languages are German 'de' and English 'en'

Output:



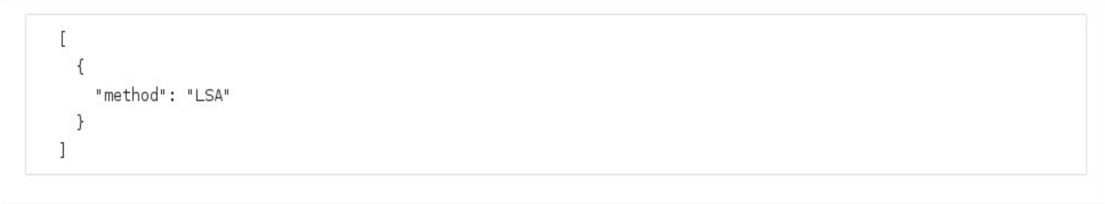
```
[
  {
    "bert_model": "bert-base-cased",
    "language": "en",
    "method": "BOC",
    "num_concepts": 5,
    "num_of_words_per_topic": 3,
    "number_of_required_clusters": [
      4
    ],
    "preprocessing": "True",
    "relevant_docs_per_cluster": 3
  }
]
```

Figure 12 Topic Detection Training Configuration Output JSON example

Get Inference configuration parameters. This function retrieves the inference/test parameters. The only required parameter for the inference stage is the method to be used for inferring the category of some given input text. If the model using the defined method in the inference parameters does not exist, an error will be shown. Possible values for the method parameter are BOC, LDA, LSA.

curl -X POST -H "Content-Type: application/json" -d '{ "method": "LSA" }' http://localhost:5000/setInferenceConfig

Output:



```
[
  {
    "method": "LSA"
  }
]
```

Figure 13 Topic Detection Inference Configuration Output

4.6 Network Analysis

Briefly, network analysis is the process of uncovering hidden patterns regarding the behavior and relations among individuals in networks through the use of a wide range of computational and statistical methods. Examples of those patterns are distribution of relations among the individuals, the underlying factors that determine the relations, and cohesive groups of individuals with dense relations, etc. The network analysis component in ROXANNE provides the functionalities for the following analysis

- **Social influence analysis**: to assign to each individual a relative importance score that measures its influence compared to that of other individuals. Individuals having highest scores are notable ones, and in the case of criminal networks, should draw more attention of the investigators
- **Community detection**: to uncover the cohesive groups of individuals whose intra-group interaction is denser and more frequent than their interaction with the rest of the network. Investigators may benefit from these uncovered groups when analysing the patterns of interaction among entities in a criminal network. While the human eye and mind can hardly identify cohesive

subgroups, the clustering of social relations is an important element of society’s (and criminal groups’) organization.

- **Link prediction**: In practice, people interact and communicate with each other through many channels that are not always observable. Hence, many interactions and relations among individuals in a network are hidden or not observed. Moreover, the network evolves over time: new interactions and relations are constantly added into the network. In this task, we aim to uncover these missing or hidden, unobserved interactions and relations in the network, as well as to predict the most probable ones to be formed in the near future.

4.6.1 Data Input

The input for this network analysis components consists of (1) a network, and (2) the desired analysis to be perform on the network. The network can be described by a JSON object that contains information a bout the list of nodes (or individuals) and edges. The desired analysis is one of the three above, i.e., social influence analysis, community detection, and link prediction

An example input that request for social influence analysis is as follows

```
{
 "task_id": "sia_xxxx",
 "task": “social_network_analysis”
 "network":
     [
     {"type": "node", "id": "Satam_Suqami", "properties": {"type": "person", "name": "Satam Suqami"}}
     ...
     type":"edge",  "source":  "Samir_Kishk",  "target":  "Essid_Sami_Ben_Khemail",  "properties":  {"type":
"other_associate","observed": true,"weight": 1}}
     ...
      ]
 "options": {"method": "authority", "parameters": {}}
}
```

Another example of input that request for link prediction is as follows:

```
{
       "job_id": "slp_xxxx",
       "task": “link_prediction”
       "network":
                [
               {"type": "node", "id": "Satam_Suqami", "properties": {"type": "person", "name":
"Satam Suqami"}}
          ...
```

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

```
          type":"edge", "source": "Samir_Kishk", "target": "Essid_Sami_Ben_Khemail",
"properties": {"type": "other_associate","observed": true,"weight": 1}}
          ...
                ]
       "options": {"method": "jaccard_coefficient", "parameters": {“sources”:["Satam
Suqami"]}}
}
```

4.6.2 Output

The output of this network analysis component is the result of the requested analysis on the input network. That is, the importance scores of individuals, the communities and their member individuals, or the predicted links when the requested analysis is social influence analysis, community detection, or link prediction respectively. These outputs are also produced in JSON format.

An example output for social influence analysis is as follows

```
{
       "task_id": "sia_xxxx",
       “result”:
       {
              “success”: 1,
              “message”: “The analysis was performed successfully”
              "scores":[{"id": "Satam_Suqami", "score": 0.3}, {"id": "Essid_Sami_Ben_Khemail",
"score": 0.2},
         {"id": "Samir_Kishk", "score": 0.1},...]
       }
}
```

Another example output for community detection is as follows

```
{
       "task_id": "sia_xxxx",
       "task": “community_detection”
       "network":
                [
               {"type": "node", "id": "Satam_Suqami", "properties": {"type": "person", "name":
"Satam Suqami"}}
          ...
          type":"edge", "source": "Samir_Kishk", "target": "Essid_Sami_Ben_Khemail",
"properties": {"type": "other_associate","observed": true,"weight": 1}}
          ...
                ]
       "options": {"method": "hierarchical", "parameters": {‘K’:5}}
}
```

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

4.6.3 Configuration Parameters

The configuration parameters in this network analysis component include the choices of analysis to perform and the method for performing the chosen analysis. More detail on the set of available methods is presented in deliverable D6.1 Preliminary report on network analysis. In the following, we give a brief overview of these methods.

Method for social influence analysis*: pagerank*, *authority*, *betweenness*, and *closeness_centrality*

Method for community detection*: k_cliques*, *modularity*, l*abel_propagation*, *spectral, and hierarchical*

Method for link prediction*: resource_allocation_index*, *jaccard_coefficient*, *adamic_adar_index*, and

For community detection, another parameter is ‘K’ which denotes the desired numbers of communities. For link prediction, another parameter is ‘source’ which is a set of source node to predict the link for.

An example configuration for requesting for social influence analysis using authority method is as follows

```
"options": {"method": "authority", "parameters": {}}
```

Another example configuration for requesting for link prediction using jaccard_coefficient method is as follows

```
"options": {"method": "jaccard_coefficient", "parameters": {“sources”:["Satam Suqami"]}}
```

4.7 Input/ Output GUI

ROXANNE platform allows end user to process files through complex processing chains. Data are then enriched with all processing results and finally displayed through advanced results GUIs.

The first version of the ROXANNE platform manages audio files. It will extended to text and videos for next field tests.

The way to process new files is the following:

- End user upload his audio files through upload GUI, using files or directory drag and drop.
- He then configure the processing to apply to these files
- After validation all files are uploaded to the server and the processing starts.

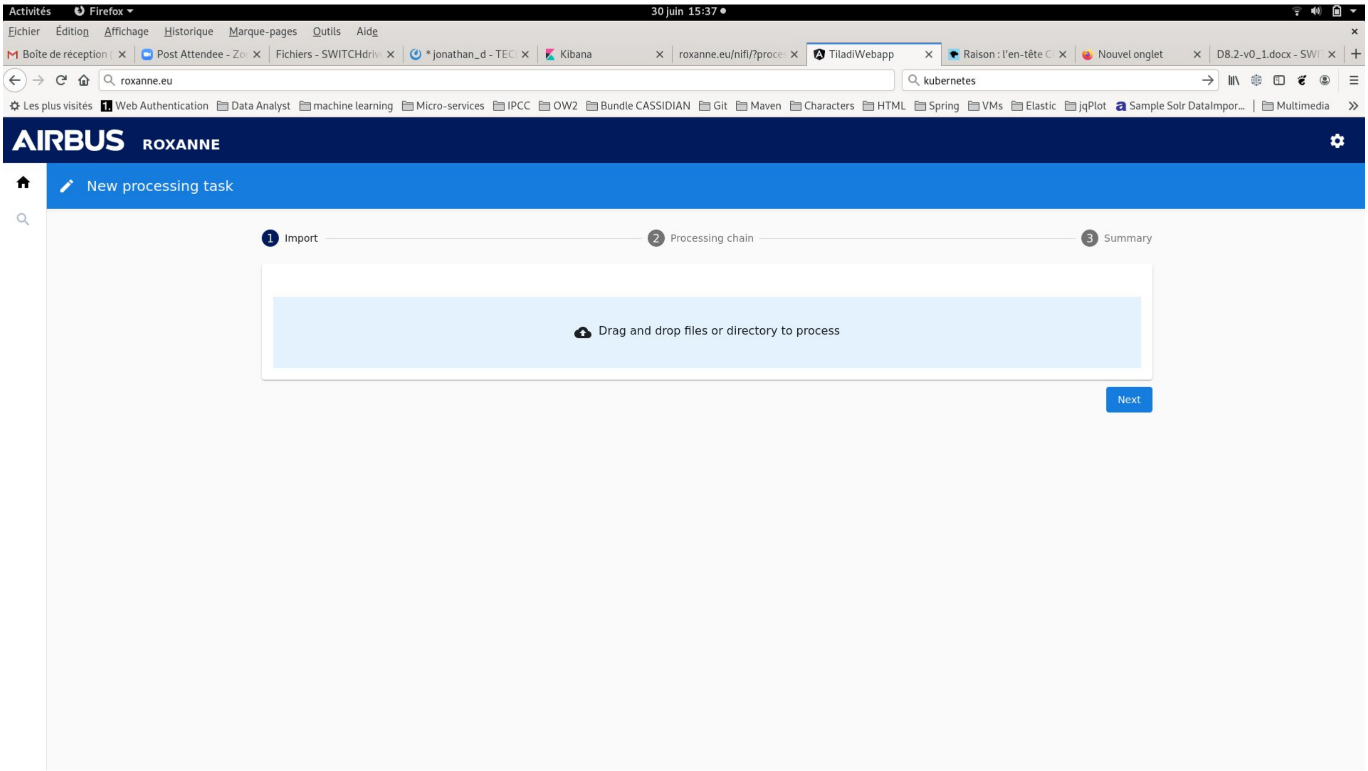
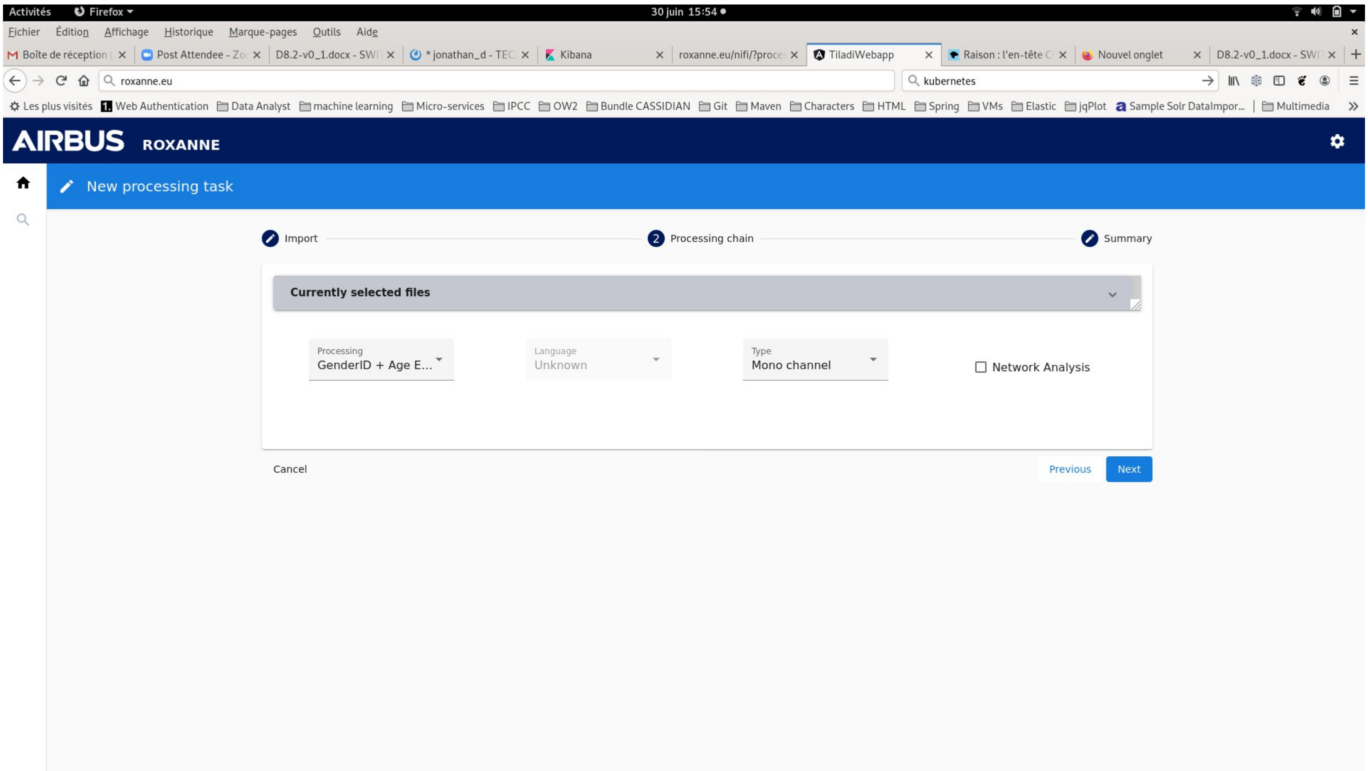


Figure 14 Files Upload GUI



Processing: GenderID + Age E...

Language: Unknown

Type: Mono channel

□ Network Analysis

Figure 15 Processing Chain Configuration GUI

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

In the first version of ROXANNE platform, end user could only upload 2 kinds of files:

- Stereo Audio files, with 1 speaker per channel. In this case, files will be split into 2 sub files, 1 per speaker and each sub file will be processed through audio components
- Mono wav file: if dataset is already split into 1 audio file per speaker, no pre-processing is done on input files. In this case, files must be named using following pattern: <conversationID>_<channelID>.wav

In the first version of ROXANNE platform only 1 audio processing chain is available, using following components:

- Voiceprint extraction
- Languageprint extraction
- Language identification
- Gender identification
- Age estimation
- Speech recognition
- Named Entity Recognition (on text extracted from SpeechRecognition)
- Topic Detection (on text extracted from SpeechRecognition)

User could then choose to run Network Analysis at the end of the processing of all files.

In this case, a network is built based on voiceprints comparison between all conversation files. This network is then analysed using network components and could be visualized through network GUIs.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 833635. No part of this document may be used, reproduced and/or disclosed in any form or by any means without the prior written permission of the ROXANNE project partners. © 2019 – All rights reserved.

5. Conclusion

The purpose of this deliverable is the collection of training materials for the modules/tools of the first field test of the ROXANNE project. Considering the end-user training requirements and the needs of the LEAs, the training material is documented in order to familiarized the non-experts to the usage of high-tech components.

Using a Web based Training approach, the trainees will be able to "asynchronously" attend online courses related with the tools of ROXANNE project.

The first volume of training material contains seven tools mainly related with audio processing with the main objective to provide the guidelines for the usage and understand the purpose of each tool. While the GUI is at an early stage, extensive description of the configuration fields, the input formats and the expected output is given.

The second volume of this series of deliverables will focus on the tools to be used for the second field test as well as to enhance the material of the reported tools based on the development progress of the ROXANNE project.