TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Jelizaveta Kärner 204702IASM

Creation of Strongly-Labelled Dataset of Tallinn Urban Noise

Master's thesis

Supervisor: Jaanus Kaugerand
PhD

Tallinn 2024

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Jelizaveta Kärner 204702IASM

Tugevalt märgendatud Tallinna linnamüra andmebaasi loomine

Magistritöö

Juhendaja: Jaanus Kaugerand
Doktor

Tallinn 2024

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Jelizaveta Kärner

02.01.2024

Abstract

According to the European Environmental Noise Directive (2002/49/EC), all European Union Member States are required to monitor environmental noise levels and their sources. This work focuses on solving a data engineering problem to create an audio dataset for the task of recognizing urban noise sources in the Tallinn city.

The study provides a review of existing urban noise datasets as well as an overview of taxonomy methods used in the industry. A two-level taxonomy of urban noise events in Tallinn is proposed, which combines the practices of the urban taxonomies based on previous studies, as well as the noise pollution classification of the Common European Noise Assessment Methods (CNOSSOS-EU). The study offers a Tallinn urban noise dataset consisting of 192,000 files with a total duration of 53.33 hours. The dataset is formed based on data collected with processing existing datasets, data search through the open platform Freesound and noise recording sessions. Analysis of the dataset validation results are offered among with suggestions for further improvement.

This thesis is written in English and is 68 pages long, including 5 chapters, 24 figures and 5 tables.

Annotatsioon

Tugevalt märgendatud Tallinna linnamüra andmebaasi loomine

Üks tänapäeva ühiskonna saastetüüpe on müra saaste. Euroopa keskkonnamüra direktiivi (2002/49/EÜ) kohaselt on kõigil Euroopa Liidu liikmesriikidel kohustus jälgida keskkonnamüra taset ja selle allikaid. Käesolev töö keskendub andmetehnika probleemi lahendamisele mis koosneb heliandmete kogumiku loomisest linnamüra allikaite tuvastamise jaoks Tallinna linnas.

Uuring pakub ülevaadet olemasolevatest linnamüra andmekogudest ning tööstuses kasutatavatest taksonoomia praktikatest. Lisaks luuakse Tallinna linnamürasündmuste kahetasandiline taksonoomia, mis koosneb 38 klassist. Pakutud lahendus ühendab varasemate uuringute põhjal välja kujunenud linnataksonoomiate praktikaid ja Euroopa Ühised Müra Hindamismeetodid (CNOSSOS-EU) mürasaaste klassifikatsiooni. Uuring pakub Tallinna linnamüra andmekogu, mis koosneb 192 000 failist kogu kestvusega 53,33 tundi. Andmekogu on moodustatud olemasolevate andmekogude töötlemise, andmete otsingu Freesound avatud platvormi kaudu ja mürasalvestussessioonide abil kogutud andmete põhjal. Andmekogu valideerimise protsessi viidi läbi koolitades väikest masinõppemudelit helituvastuse jaoks. Koolitatud mudelit hinnati kaotusega 0,673 ja täpsusega 78,1%.

Tallinna linnamüra mittesünteetilist andmekogu sisaldab muutumatuid helifaile koos augmenteeritud andmetega, jaotusega 47% ehk 25 tundi esimese ning 53% või 28,28 tundi teise jaoks. Originaal- ja augmenteeritud andmete suhe võib olla aluseks edaspidi uuringuteks, et analüüsida, kuidas augmenteeritud andmete hulk mõjutab masinõppe mudeli täpsust.

Lõputöö on kirjutatud Inglise keeles ning sisaldab teksti 68 leheküljel, 5 peatükki, 24 joonist, 5 tabelit.

List of abbreviations and terms

NN	Neural Network
ISC2PT II	Intelligent Smart City and Critical Infrastructure Protection Technologies
UART	Universal Asynchronous Receiver-Transmitter
SER	Sound event recognition
CNOSSOS-EU	Common Noise Assessment Methods in Europe

Table of contents

List of figures

List of tables

1 Introduction

One of the types of pollution that society faces today is noise pollution. Noise pollution refers to any unwanted or disturbing sound that disrupts the normal functioning of humans and animals. It is usually caused by human activities such as transportation, construction, industrial activities, and recreational activities. Exposure to excessive noise levels can lead to various health problems such as hearing loss, high blood pressure, sleep disturbance, and stress [1].

In an effort to combat noise pollution, Europe has taken action by implementing the Environmental Noise Directive (2002/49/EC) [2]. The directive mandates European Union (EU) Member States to assess the level of exposure to environmental noise through strategic noise mapping using Common Noise Assessment Methods in Europe (CNOSSOS-EU) and to create action plans aimed at reducing noise pollution [3]. Since June 2007, all EU countries are required to produce strategic noise maps for major roads, railways, airports, and urban areas every five years. National competent authorities utilize these noise maps to identify areas that require immediate action, while the European Commission utilizes them to gain a comprehensive understanding of noise exposure throughout the EU [2].

As part of the Environmental Noise Directive, the Estonian Land Department has created a map application with information on the average level of noise pollution [4]. To date, the numerical values of noise indicators are obtained by calculation using mathematical models of the propagation of noise of various types in landscape areas. The results of calculations of noise from automobile, tram, rail and air transport are based on average statistical data on the frequency of traffic, permitted speed indicators and the nature of the landscape and the degree of sound reflection from surrounding surfaces [5]. For example, when modelling railway noise, data on the railway transport schedule, type of train, number of cars and noise coefficients specified in CNOSSOS-EU are used. To calculate construction noise, the mathematical model considers the nature of the noise, which is divided into three types depending on the direction of propagation of construction noise in the area:

- point source of noise,
- linear noise source,
- area [3].

Data from point measurements of noise levels at different distances from the noise source are also used to determine the strength of industrial noise propagation [6].

Recently, two large projects that are related to noise measurements within city limits of Tallinn have been carried out – Networking Technologies (SmENeTe) [7] and Intelligent Smart City and Critical Infrastructure Protection Technologies (ISC2PT) [8]. The former, with the aim of deploying a large network of ca 900 sensors to the urban environment ended in 2019. Sensor nodes are placed on street lighting posts of Tallinn and are connected to a network. Different combinations of sensors can measure noise level and sound radars can analyse traffic conditions [9]. The latter ended in June 2022 with the aim of making sensors even smarten by updating their software by using ideas from the field of artificial intelligence. For example, the ISC2PT project developed methods to differentiate between different types of noise which can later be used to create a noise pollution map of the city of Tallinn and make today's process more automated.

This task can be carried out using artificial intelligence and more specifically machine learning algorithms embedded into the microcontroller. However, before working with artificial intelligence solutions, it is necessary to solve the problem of data on the base of which these solutions can be developed. A set of practices and methods for collecting, processing, cleaning, and preparing data for further analysis, research, and artificial intelligence development is called data engineering. This work focuses on solving the data engineering problem of creating an audio dataset for the urban noise source recognition task.

1.1 Research problem

The dataset's creation is based on determining the noise structure within the urban environment. Therefore, the definition of noise in the urban space of the Tallinn city is identified as one of the research problems of this thesis. The choice of categories for different noise sources, the definition of their name and categorization into a hierarchical structure is called the noise taxonomy problem. Taxonomic solution requires to be compatible with existing practices and correspond to certain regulations within the European Union on which categories of noise pollution should be monitored [3].

Collecting equal in quality audio data for domain-specific tasks is a major challenge. Due to high financial and technical requirements there are very few complete and publicly available audio datasets sufficient in volume for further work while also recorded in the context of the real world. As a result, synthetic data generation is often applied using software libraries such as Scaper [10] and Pyroomacoustics [11] to meet the training requirements of machine learning systems. While these software libraries might be useful in the experimentation and development of many machine learning audio systems, there is an opinion that synthetic data can never fully reproduce the nuances of sounds recorded in a real acoustic environment [12]. Therefore, this work is focused on creating a dataset based on real world data.

There are different ways to approach the problem of collecting real world data of urban noise, each with varying levels of complexity. For example, there are a number of studies that provide ready-made datasets of non-synthetic noise in the urban environment of different cities [13], [14], [15], [16], [17], [18], [19]. The release of datasets under a Creative Commons license makes it possible to consider them as one of the data sources. However, these datasets deviate to some extent from the CNOSSOS-EU [3], limiting their complete utility.

Another collection method is to conduct noise recording sessions in the real world. This method guarantees the most predictable result since it makes it possible to control sound quality through the choice of equipment and the choice of location for recording. At the same time, it requires the most human involvement since it takes a large amount of monotonous work, which is difficult to automate at the following stages:

- checking the recorded material for compliance,
- cleaning the material from unwanted noise,
- sorting material by noise sources.

Another way is collecting data from small publicly available datasets using large platform where anyone can share a free dataset – Freesound [20]. This is a platform for openly available audio datasets released under Creative Commons licenses with allowance of correction and reusage of released datasets. The collaborative approach of the site where each user can leave notes and feedback on the dataset, as well as offer their own changes, has contributed to the development of the community [21].

Despite the impressive number of open audio datasets, the lack of standards in describing technical characteristics and the process of creating and processing audio material has led to a situation where users are free to describe the published materials at their discretion. Therefore, data validation and audio standardization methods should be taken into account when compiling an audio dataset of sufficient size for the task of recognizing a noise source from small open datasets from Freesound platform.

Thus, the second research problem of the thesis is to study such methods of collecting real world audio data of urban noise like processing existing datasets, data search through the open platform Freesound and noise recording sessions.

1.2 Method

The following methods are used in this thesis to develop a data engineering solution for creating an audio dataset for the urban noise source recognition task:

- creating an overview of existing audio datasets containing urban noise and methods for their gathering,
- overview of existing urban noise taxonomy practices,
- development of a taxonomy of urban noise that is relevant for the Tallinn city, compatible with existing taxonomies and has the potential for further use,
- describing methods for gathering audio data of urban noise and creation of audio dataset of urban noise for Tallinn city,
- validation of created dataset.

1.3 Overview of the thesis

The description and analysis of existing datasets containing urban noise can be found in the section 2. This section also contains a description of the main characteristics and terms applied to audio datasets.

A review and comparison of taxonomic approaches can be found in section 3. This section examines both the taxonomy used to create the datasets described in section 2 and the noise classification standard adopted in the European Union. In the last part of the section 3, the author proposes a taxonomic system of noise for the Tallinn city.

Finally, Section 4 focuses on the Tallinn urban noise dataset. The section includes a description of collecting audio processing data using existing datasets, data units and noise recording sessions. The concept of data balancing is also covered. In addition, the section proposes a method for validating the created dataset based on a machine learning model for speech recognition. In conclusion, a description of the dataset, an analysis of the validation results and suggestions for improvement are offered.

2 Existing Urban Sound Datasets

For a general understanding of the possibilities of using existing non-synthetic or real-world audio datasets, it is necessary to consider and describe the most significant ones available in the public domain. The main characteristics for comparison between datasets are selected:

- data source,
- total duration of the dataset in hours,
- total number of sound files,
- length of one file in seconds,
- the number of classes represented in the dataset,
- labelling type.

A sound that can be identified within a single audio file is called a sound event. A sound event can be any sound source, and even its absence. In this case, the sound event can be labelled "silence". The process of identifying a sound event and assigning a label to it is called annotation and can be done using variety of approaches.

It is assumed that each sound event within one audio file has a beginning named onset and an end – offset. In some cases, the onset and offset of an event may coincide with the start and end of the audio file itself. Specifying the onset and offset of the sound event as an additional parameter within the data annotation process is considered to be "strong annotation". On the contrary, "weak annotation" is when the onset and offset are not specified in the audio dataset and the sound event lasts less than the duration of the audio file. In this case, the annotation only indicates the presence of a certain sound class but does not indicate when exactly it occurs [22], [23].

However, there are cases where onset and offset are not specified as separate characteristics, but the length of the sound event is either near or equal to the length of the sound file. In such cases, the author noted a certain divergence of opinion as to whether the annotation is strong [24] or weak. Thus, the description of the same dataset in different works may differ depending on the interpretation of the study. For example, the UrbanSound8k, which will be described in more detail below, is referred to as both a weak [12] and a strong annotation dataset [25]. In other work related to the compilation of the dataset, it is also stated that part of the dataset can be considered strong since it contains files with only one label per file marked as “Present and predominant” [13]. Based on the works of Serra *et al* [13] and Adavanne *et al* [24], the author decided to follow the approach where the annotation is considered as strong when specifying onset and offset as a separate indicator. Similarly, annotation is strong for the cases when the length of the sound event is near or equal to the length of the sound file.

Another characteristic of an audio dataset is the number of audio events per audio file and whether different audio events overlap each other at the same time. Among the sources studied there is no consensus on what terms to use in identifying this characteristic, so the author of this thesis uses the approach proposed in work by Ooi *et al* [12]. The principle is to divide datasets into two categories:

- monophonic where only one sound event occurs in all audio files,
- polyphonic where the number of sound events per class can be more than one and different events can overlap.

Below is a description of the most significant natural (non-synthetic) datasets representing the sounds of urban space. In this section, the author focuses on the description of datasets, their formats, main characteristics, and scope, without focusing on the issue of taxonomy or classification, as this topic will be developed in more detail in the next section.

2.1 Freesound FSD50k Dataset

The FSD50k (Freesound Dataset 50k) dataset is a large collection of annotated audio recordings from Freesound platform for sound recognition and classification research. It consists of 51,197 audio files (as of 10.07.2023), with length ranging from 0.3 to 30 sec, covering a wide variety of sound events, including musical instruments, human activities, and sounds of nature. The dataset is annotated with 200 classes of sound events (as of

10.07.2023), making it one of the largest and most diverse audio datasets available. The FSD50k dataset is intended for research purposes and is freely available for download from the Freesound website under a Creative Commons Attribution license [13].

A larger dataset than the FSD50k is the AudioSet dataset based on over 2M tracks from YouTube videos and encompassing over 500 sound classes [17]. A limitation of this AudioSet is that the YouTube videos which the dataset is based on, are subject to the YouTube Terms of Service. Thus, the use of AudioSet can be somewhat difficult. In addition, the storage of instances is entirely up to YouTube users and can be deleted at any time. FSD50k was created as an open alternative to AudioSet and based on the taxonomy developed by AudioSet.

The dataset annotation was done manually using the Freesound Annotator, where annotators marked the presence of a particular class in a particular audio file [26]. At the same time, part of the dataset contains more than one class per sound file, while files from other part are no longer than 4 seconds long and contain only one class. In general, the authors of the dataset themselves designate the labelling approach as weak labelling.

Since the FreeSound platform allows the reuse of existing sound datasets for creation a new ones, some of the possible problems to consider when working with data from the FreeSound platform is the risk of repetition of the same sound instances in several datasets. The FSD50K dataset authors also tried to avoid duplicate clips by dividing the dataset onto dev and eval split. The audio clips have been grouped within each split to retain original files from the same uploader. As a result, eval is exhaustively labelled, while annotations in dev are correct but potentially incomplete.

Thus, the FSD50K is a fairly large open sound dataset, but it has a certain diversity and heterogeneity in sound characteristics that must be taken into account when working with this dataset. Even if the total number of sound instances is 51,197, the authors themselves warn about the possibility of having 80% incomplete annotations (40,966 instances as of 10.07.2023) [13].

2.2 UrbanSound and UrbanSound8k Datasets

UrbanSound is a dataset specializing mainly on sound classes that are typical for the urban environment. The dataset was created by researchers at the Music and Audio Research

Laboratory and the research group from the Center for Urban Science and Progress at the New York University. It consists of 1302 full length audio recordings with duration of audio files from 1 second to 599.4 seconds. The dataset includes ten different urban sound classes: air conditioner, car horn, children playing, dog bark, drilling, engine idling, gunshot, jackhammer, siren, and street music. The total volume of the dataset is 27 hours of real field-recorded urban environment audio which were collected from Freesound platform [14].

The annotation process was carried out manually and included the start and end of the sound event as well as its class designation within the audio file. The authors also designated such a characteristic as a salience. This indicator shows whether the event was subjectively perceived by the annotator as a foreground or background event. As a result, 3075 labelled events were noted in 1302 audio files with a total length of 18.5 hours. Comparing these values, it becomes clear that there can be several events per sound file and thus considered by author of this thesis as polyphonic.

UrbanSound8k is a monophonic subset of the UrbanSound dataset, consisting of 8732 audio recordings with a duration of up to 4 seconds each that have been manually annotated and verified with a total volume of 8,75 hours. This subset does not specify onset and offset as separate characteristics but assumes that the length of the event is approximately equal to the length of the audio file [14].

One of the strengths of the UrbanSound8k dataset is its diversity of urban sound classes, which enables researchers to test the robustness and generalizability of machine learning models [14]. However, since the dataset was compiled based on different datasets from Freesound, it must be taken into account that files can have different sampling rate and audio is recorded at different bit depths [12].

2.3 SONYC Urban Sound Tagging Datasets

The Sound of New York City (SONYC) is a research project to create an intelligent, sensor-based system for monitoring, analysing, and mitigating urban noise pollution in New York City. SONYC uses a network of sensors deployed throughout New York City to monitor and record noise levels and sound events in real time. The data collected by these sensors is then analyzed using machine learning algorithms to identify sources of noise pollution and track their impact on city residents [15].

SONYC-UST-V2 and SONYC-UST-V1 are two versions of the Urban Sound Tagging (UST) dataset which were developed within the SONYC project and belong to a collection of audio recordings of environmental sounds in New York City urban settings [18], [19]. Since the entire cycle of creating these datasets from collecting sound files to annotation and validation was carried out within the framework of one project, all files have similar sound recording frequency and quality because the same microphone sensors were used.

Sound was collected through the acoustic sensors' network consisting of more than 60 sensors across the New York City. The main concentration of sensors is located in the Manhattan area, but areas such as Brooklyn and Queens are also covered. The range of sound that such a module can capture is 32-120 dBA. Each of the sensors is located at the height of 4.5 to 6.6 meters from the ground and records sound segments 10 seconds long with a random interval between recordings.

The creators of the dataset followed a weak labelling approach for the annotation and dataset was mainly produced using a crowdsourced approach with volunteers for manual annotation. The dataset is considered as a polyphonic. The dataset also has the “SONYC-team-verified” mark for those sound files that were cross-checked by the team members of SONYC project. The first version of the dataset consisted of 3086 sound files, each 10 seconds long. The authors also proposed a distribution into training, validation, and test splits (2351 / 443 / 274 recordings respectively) where the annotation of the validation and test splits have the “SONYC-team-verified” note.

In the original version of SONYC-UST-V1, the dataset consists of audio annotation files, but later, starting from the second version (SONYC-UST-V2), the location of the sound sensor and time when sound was recorded (the spatiotemporal context) is also taken into account. Another difference between SONYC-UST-V2 and SONYC-UST-V1 is the size and diversity of the dataset. SONYC-UST-V2 is a much larger dataset than SONYC-UST-V1, containing over 18,510 audio recordings of urban sounds. Out of all the recordings, 1380 have confirmed annotations. These annotations consist of 716 recordings from the SONYC-UST-V1 test and validation sets, and 664 new recordings that form the SONYC-UST-V2 test set.

2.4 SINGA:PURA (SINGApore: Polyphonic Urban Audio)

The SINGA:PURA dataset was created within the project to identify and mitigate noise pollution in Singapore City. The dataset includes strongly-labelled recordings of urban sounds collected from 14 acoustic sensors distributed throughout Singapore. The sound was collected by recording sounds from August 3, 2020, to October 31, 2020, using two types of microphones: single-channel Knowles SPH0645 microphones and seven-channel MiniDSP UMA-8 v2 microphone arrays, both controlled by Raspberry Pi 3 Model B. The recordings were made at a 44.1 kHz sampling rate, and each audio file contains information on its spatiotemporal context [12].

There are 6,547 strongly-labelled manually annotated recordings, each with a duration of 10 seconds, and are organized by the date of recording in separate folders. The members of the project manually annotated these recordings, ensuring that the time of start and end points of each event for a particular class were correctly matched. In cases where the sound was difficult to identify or unclear, the subjective annotation was used. The annotation approach also included three different proximity ranges for the sound source – "near", "far" and "moving" – based on the annotator's perceptions. Although the annotation is strongly-labelled, the dataset is considered as a polyphonic [16].

In addition to the annotated recordings, there are also openly available 72,406 unlabelled polynomic recordings. Both annotated and unlabelled splits are licensed under the Creative Commons Attribution-ShareAlike 4.0 International license, which allows for its use and distribution with appropriate attribution and under the same license.

2.5 Summary

In this part of the research work, six publicly available audio datasets were considered that consisted only of recordings from the real world and did not contain synthetic audio. Some of the datasets include a larger number of classes with a wider range of sound variations, and one has only 10 classes, but in each of the datasets include classes associated with urban noise.

In Table 1, one can find the comparative characteristics of the audio datasets described previously. Of the 6 datasets, 5 specialize in urban noise and 1 FSD50k is a wider spectrum dataset that does not focus only on the urban environment. At the same time, as can be seen from Table 1, FSD50k is the dataset with longest duration out of all considered. The total amount of annotated data is 108 hours divided into 51,197 clips ranging in length from 0.3 to 30 seconds. In second place in terms of hours is the SONYC-UST-V2 with 51.4 hours of audio data divided into 18,510 clips of 10 seconds each. In third place is the UrbanSound dataset where the total number of hours is 27 split into 1302 clips ranging in length from 1 second to 30 seconds.

It is noteworthy that the only monophonic dataset is at the same time practically the shortest in terms of time (8.8 h) – UrbanSound8k. This dataset is also characterized by the smallest maximum duration of one clip – 4 seconds.

Table 1. Comparison of non-synthetic datasets containing urban classes.

Dataset	**Source**	**Duration (h)**	**Nr of clips**	**Clip length**	**Nr of classes**	**Label Type**
FSD50k	Freesound	108.0	51197	0.3–30s	200	Weak-labelling, polyphonic
UrbanSound	Freesound	27.0	1302	1–599.4s	10	Strongly-labelled polyphonic
UrbanSound8k	Freesound	8.8	8732	4 ≥	10	Strongly-labelled monophonic
SONYC-UST-V1	SONYC	8.6	3086	10 s	23	Weak-labelling, polyphonic
SONYC-UST-V2	SONYC	51.4	18510	10 s	23	Weak-labelling, polyphonic
SINGA:PURA	Singapore	18.2	6547	10 s	40	Strongly-labelled polyphonic

In the process of analyzing datasets, general principles for storing audio and metadata were also identified, which the author of the work will also adhere to. These principles can be considered as an industry standard. Thus, all dataset audio files are stored with the Waveform Audio File Format (WAV) extension and packed into folders.

In addition each dataset except UrbanSound uses its own scale to evaluate the degree of prevalence of a certain sound event in the file. One can find the sound event presence scale at Table 2.

Table 2. Comparison of used notations for the class presence value in the audio clip by each dataset.

Dataset name	**Possible values at the class presence scale**	**Comments**
FSD50k	PP / PNP / NP	PP – present and predominant, PNP – present but not predominant NP – not present
UrbanSound	–	–
SONYC-UST (V1 & V2)	Near / far / not sure / -1	“-1” means the proximity was not annotated
SINGA:PURA	Near / far / moving	\`near\` and \`far\` are used for stationary objects

All additional data about each file that the dataset creators consider necessary to designate are stored in a separate file in the Coma Separated Value (CSV) extension. Such data can be a file name, day and time of recording, file source (third-party dataset or sensor identifier (ID) from which the recording was made), sound recording location, enumeration of classes, start and end of the event, annotator ID, the degree of prevalence of a certain class in the file.

3 Label taxonomy

The question of the taxonomic system is an integral part of the creation of an audio urban space dataset, since it determines the order in which classes are categorized for further study. Although at first glance it may seem that in order to study the sound of urban space, it is enough to single out several generalized categories using the principle of the main sources of noise in urban space. Examples of such categories might be human noise, road noise and construction noise. However, a closer look at such a simple class system approach will raise questions about the sound categorization and classes like dogs barking or the sound of seagulls. In addition, there are forces of nature such as wind or rain that will also be picked up by the microphone and interfere with the identification of the main source of noise among the three classes. Moreover, there are sound sources that can be found both in the road space and during construction work. This is especially true for large construction equipment such as tractors, hoists or trucks with large engines that create similar sound event and can be found in both cases.

Below, the main trends in taxonomic approaches in the industry will be considered using the example of datasets that were described in the previous chapter to further form a taxonomy that is relevant for the urban space of the Tallinn city.

3.1 Taxonomy practice for audio datasets with tree structure

Among the previously described datasets, two different approaches to compiling a taxonomy can be distinguished. In the first case, the nature of the sound source is used to create a tree-like class inheritance structure. Classes in this case are distributed vertically from more general ones like "Sounds of Nature" to more specific ones like "Bee Buzzing". At the same time, on the vertical axis between the classes "Sounds of nature" and " Bee Buzzing " there can be several more generalizing classes such as "Living creatures" and "Insects". When describing the structure of a tree, classes are usually called nodes, and classes that are at the very bottom along the horizontal axis can be called a leaf node. The number of nodes between the general class itself and the leaf node along the horizontal axis is called the structure depth. In the case of taxonomies specializing on the nature of the sound source, the structure depth can be up to 6 nodes.

For example, FSDK50 dataset (200 classes) was created based on the taxonomy proposed by AudioSet [17]. The AudioSet taxonomy can be described as a layered class hierarchy structure of 632 classes with a maximum depth of 6 nodes. Thus, at the beginning of the hierarchy there are such classes as: "Human sounds", "Animal", "Music", "Natural sounds", "Sounds of things", "Source-ambiguous sounds", "Channel, environment and background"; and the lowest level classes are such as: "Wheeze", "Acoustic guitar", "Soprano saxophone", "Wind chime", "Car alarm", etc. One can find a visualization of the AudioSet general taxonomy structure in Figure 1 where the classes used in the FSDK50 dataset are also marked in red [27].

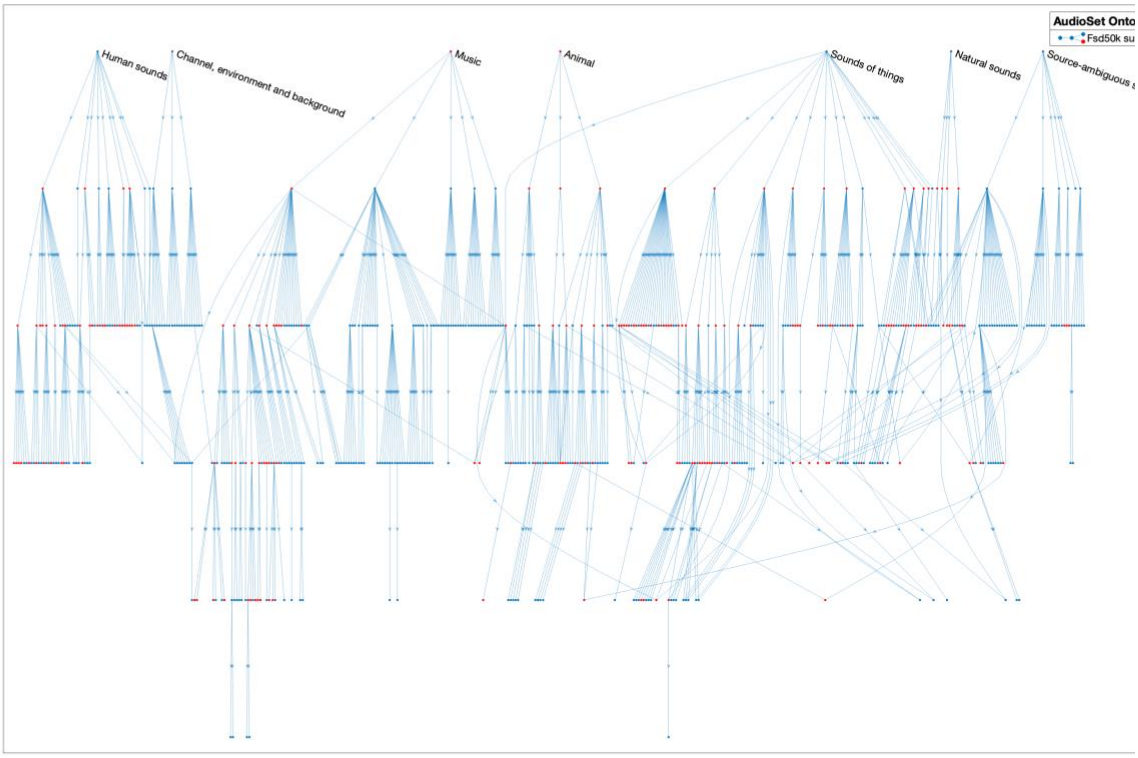

Figure 1. Structure of AudioSet taxonomy (632 classes) with matched FSD50K subset (200 classes) by red colour.

The purpose of the demonstration of Figure 1 is to visualize the general view of the AudioSet structure of the taxonomy and the relationship between different branches of the structure – chains of nodes extending from the topmost nodes. A more detailed structure within one branch using the example of the "Channel, environment and background" branch can be seen in Figure 2.

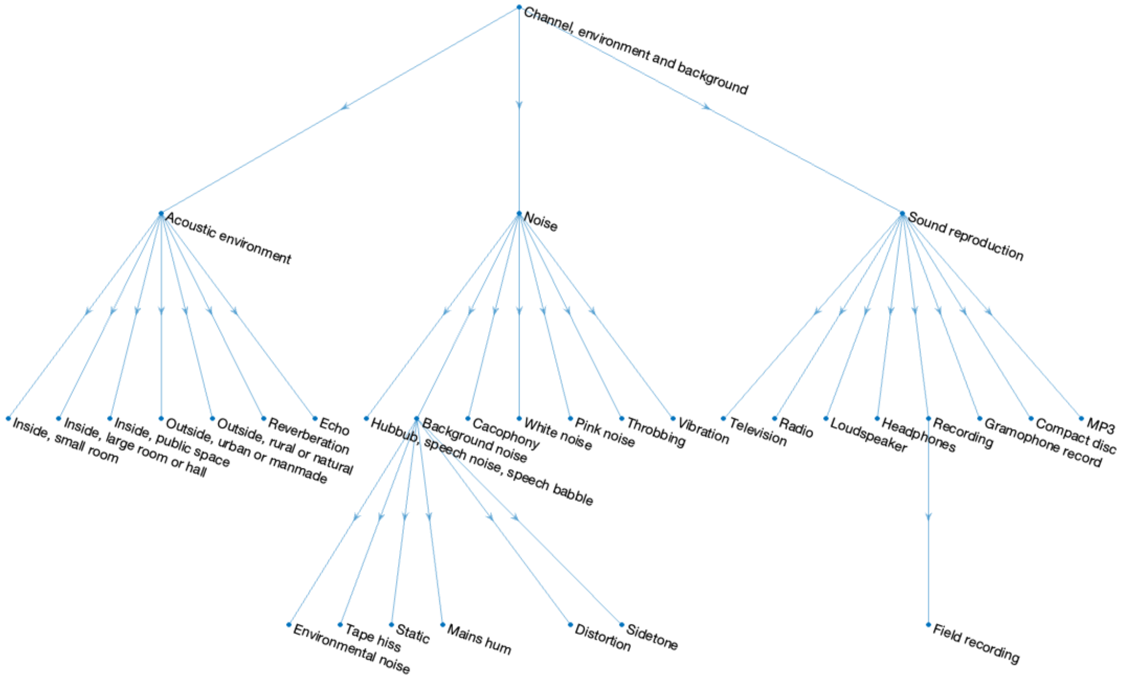

Figure 2. Hierarchical distribution of classes within the "Channel, environment and background" branch.

Since the main task of AudioSet taxonomic system is a universal audio event recognizer, the structure also tries to cover as many variations of sound sources as possible and is not limited to sounds characteristic only of the urban environment. The general list of all 632 classes is given in the JSON format, where each class has a unique ID, class name, short description and class children (if any) [28].

Another example of the similar taxonomy approach, where a lot of attention is paid to the ontology of each sound, is the Urban Sound Taxonomy [14]. In this case, several intermediate links can be combined with each other into the same leaf nodes. For example, “private car”, “police car”, and “taxi” intermediate links would be from the parent class leading to the “running engine” leaf node.

Focus on the nature of the sound source leads to endless new class refinement. To address that UrbanSound proposes the following principles for taxonomy composition:

1. consider the proposal of the earlier taxonomy system and previous studies,
2. focus on detailing the sound source at the level of low-level classes and use for example "car horn" (instead of "transport") and "jackhammer" (rather than "construction"),

3. focus initially on the classification of sounds that have a direct impact on the noise pollution of the urban environment and can contribute to further research on urban sound [14].

In part 2.2, it was described that the UrbanSound dataset consists of 10 classes (air conditioner, car horn, children playing, dog bark, drilling, engine idling, gun shot, jackhammer, siren and street music). However, the taxonomy proposed in the study is much wider and is a six-tiered class system with 53 classes at end node level. One can find the "Urban Sound Taxonomy" visualization in Figure 3 where rounded boxes denote high-level semantic classes and rectangular boxes refer to specific noise source classes. Just like the AudioSet, "Urban Sound Taxonomy" has a tree structure of nodes. "Urban Sound Taxonomy" 4 top level groups defined as “nature”, “mechanical”, “music” and “human”. Just like the AudioSet, "Urban Sound Taxonomy" also has some nodes with more than one parent. But compared to the AudioSet, nodes of different branches do not overlap. For example, nodes originating from the "Music" branch do not have a common parent with any of the nodes originating from the "Mechanical" branch. With that the structure is more predictable and understandable.

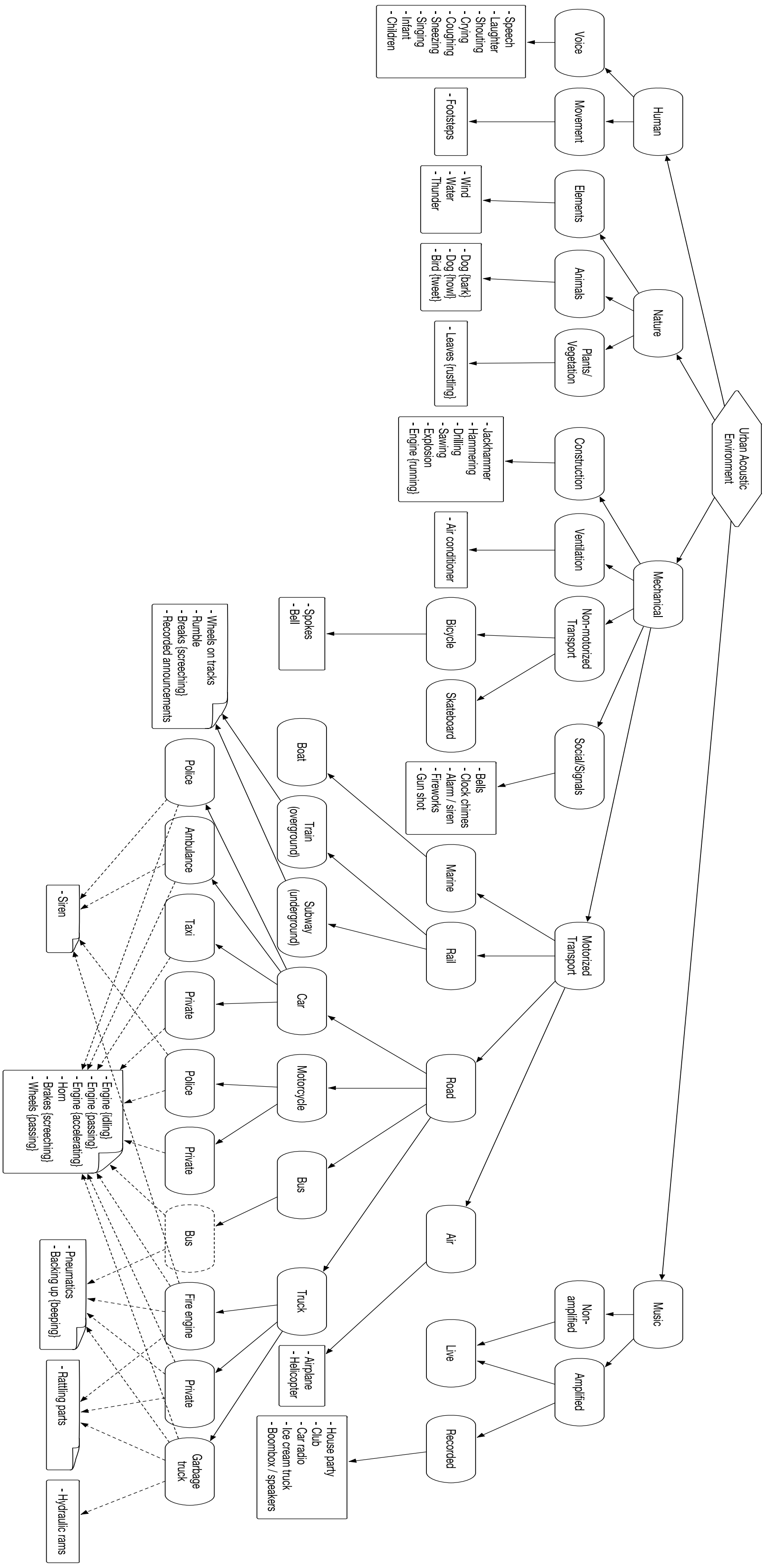

Figure 3. Urban Sound Taxonomy by UrbanSound dataset authors.

It is noteworthy that in the diagram above, some classes are the end nodes, despite being marked as a semantic class. Thus, the classes "Live" (live music), "Boat" and "Skateboard" do not have classes denoting a specific sound source. This can be explained by the opinion of the authors that the number of audio classes in urban space is potentially infinite and requires further replenishment and gives users a semantic vision, but in the current version does not carry a large practical load. A somewhat significant shortcoming is the complete absence of a description for the final classes. Although their names themselves are descriptive, it can be quite problematic to distinguish the difference between a running engine at construction work and the sound of an engine that the authors attribute to road sounds.

3.2 Two-level taxonomy practice for audio datasets

If for the first approach of constructing a taxonomy it is important to show the semantic nature of the sound source, then the second approach focuses on the ease of perception and restrictions on the structure depth to two levels. The top generalizing class in the hierarchy is called coarse-levels and the lower classes (end nodes) in this approach are usually called fine-level classes. The benefits of this approach result in a clearer structure and are better understandable within the crowdsourced annotation process [18].

An example of this approach is the hierarchical taxonomy of the SONYC Urban Sound Tagging (SONYC-UST) dataset which is shown in Figure 4. The first level consists of 8 coarse-levels that describe the attribute and how they belong to the second-level sound category. In total there are 23 categories. For example, the coarse-level “Machinery Impact” class consists of four fine-level categories: “rock-drill”, “jackhammer”, “hoe ram”, and “pile driver”. It is noteworthy that the “Powered Saw” is not included in the “Machinery Impact” class but is allocated as a separate coarse-level with three different fine-level classes: “chainsaw”, “small or medium rotating saw”, “large rotating saw”. There is also a “non-mechanical impact” category which is both coarse-level and fine-level at the same time [18].

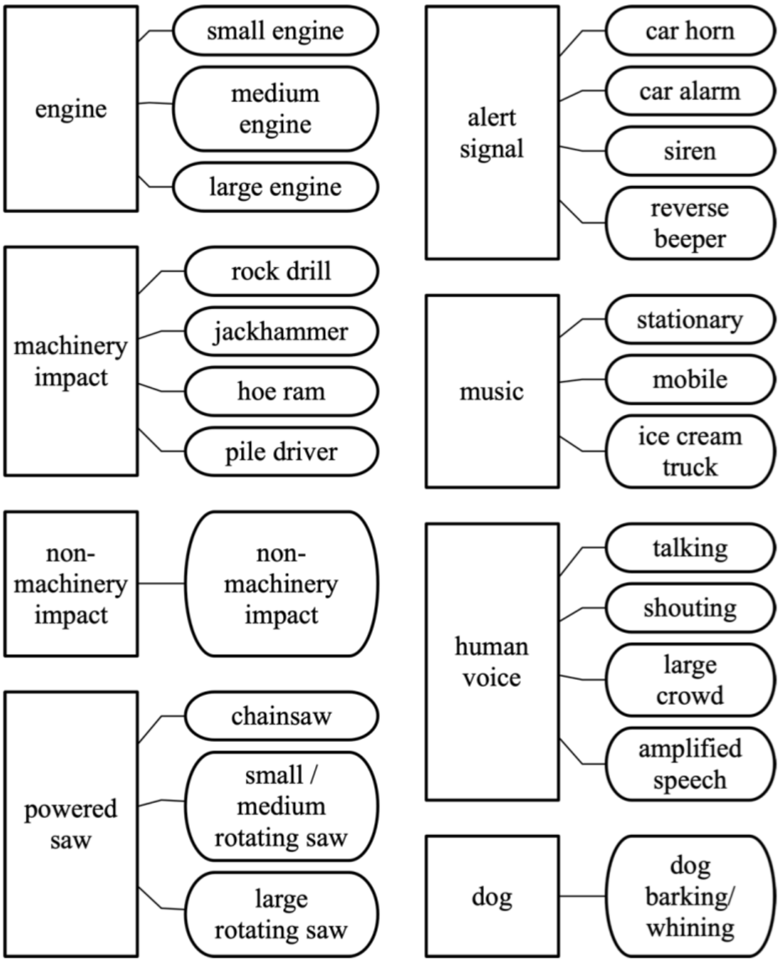

Figure 4. Hierarchical taxonomy of the SONYC Urban Sound Tagging (SONYC-UST) dataset.

The two-level taxonomy approach described above was also adopted by the authors of the polyphonic urban sound dataset with spatiotemporal context – SINGA:PURA [16]. The content of the classes has been revised to better suit the noise context of the Singapore city. The final SINGA:PURA taxonomy consists of 14 coarse-levels and 40 fine-levels classes and is shown in Figure 5. Classes that are present in both SINGA:PURA and SONYC Urban Sound Tagging taxonomies are marked in green. New classes added by the authors of SINGA:PURA are marked in red and in blue is the only class from the SONYC class list that was not included in the SINGA:PURA taxonomy – ice cream truck music.

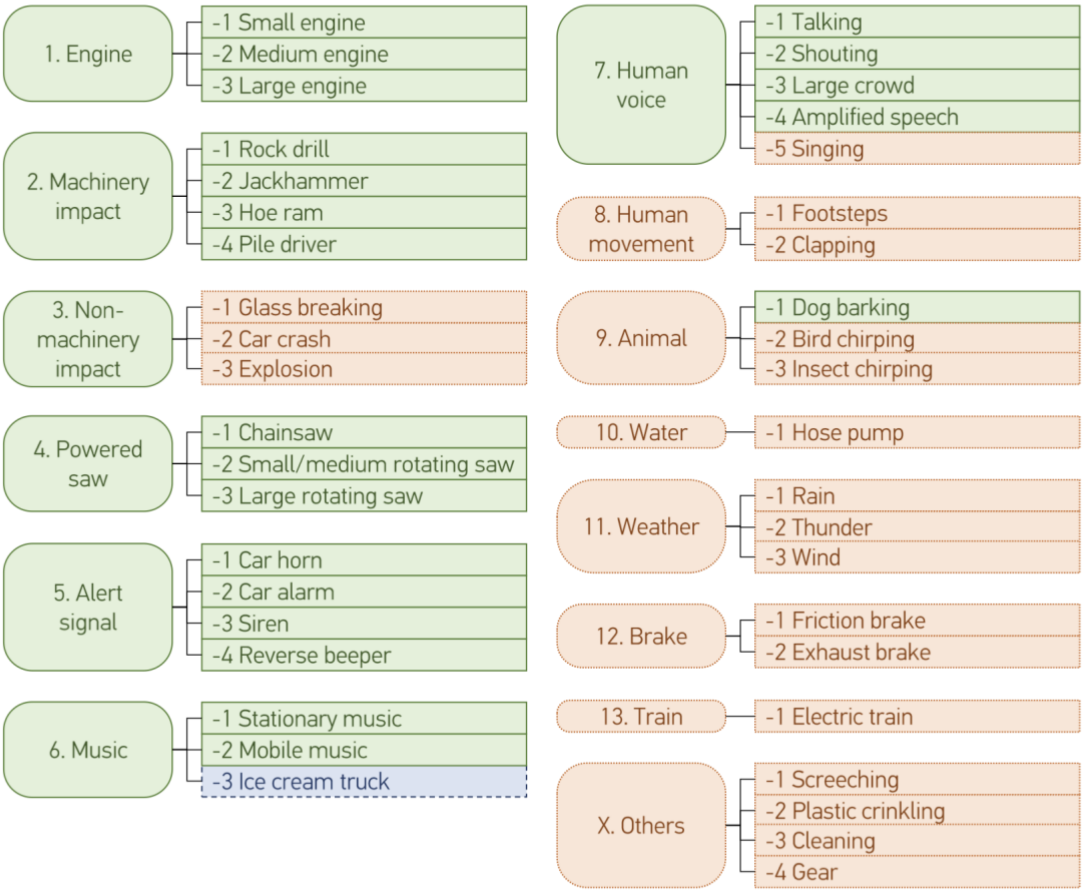

Figure 5. Label taxonomy for the SINGA:PURA dataset.

Thus, the number of fine-level classes under coarse-level "Machinery impact" was expanded: "Glass breaking", "Car crash" and "Explosion" were added. Human movements were brought into a separate fine-level class, which included footsteps and clapping; coarse-level "dog" was expanded to "animal" under which the fine-levels "Bird chirping" and "Insect chirping" were defined; a new coarse-level "water" was added in which there is one fine-level class "Hose pump". In addition, a coarse-level "train" was added with the only fine-level class "electric train"; Brake sounds have also been separated into a separate "brake" category with two finer classes "Friction brake" and "Exhaust brake". As well as in Urban Sound Taxonomy, in a separate category "Weather" the sounds caused by such weather phenomena as rain, thunderstorm and wind are included. There is also a coarse-level class "others" which includes: "Screeching", "Plastic crinkling", "Cleaning", "Gear". Similar to the previous taxonomies, there is no class description, so it can be difficult to understand on what basis the separation between "Non-machinery impact" and "Others" coarse-levels was made and what kind of noise is meant for the "Others" class.

3.3 Noise classification standards in European Union

In the EU, there are certain directives (like Directive 2002/49/EC of the European Parliament [2]) under which Member States must monitor the level of noise pollution, draw up maps of noise pollution and develop a strategy to control the spread of noise. It is assumed that the maps are compiled based on partial measurements and subsequent calculations using mathematical models, considering the density of a particular noise, for which detailed recommendations are given in CNOSSOS-EU methodology for calculation [3]. Within CNOSSOS-EU, the following categories of noise pollution are defined for countries to monitor: road noise, railway noise, aircraft noise, construction noise. For aircraft the directive describes only methods for calculating noise prediction, and for construction recommendations are given on the methodology of sound measurements with subsequent calculation of the total background noise for a particular area. The road and railway categories are defined as a combination of different sources of vehicles with the designation of their categories.

According to the Common noise assessment methods in Europe (CNOSSOS-EU) road noise is divided into 4 sub-categories depending on the type of vehicle:

- Sub-category 1: Light motor vehicles
- Sub-category 2: Medium heavy vehicles
- Sub-category 3: Heavy vehicles
- Sub-category 4: Powered two-wheelers

Light motor vehicles are defined as passenger cars, delivery vans up to 3.5 tons, sport utility vehicles, multi-purpose vehicles including trailers and caravans (corresponds to M1 and N1 Vehicle category in EU). Medium-heavy vehicles include vans with a load capacity of 3.5 tons and above, buses, tourist cars and others with two axles and dual tires on the rear axle. This category is also referred to as M2, M3 and N2, N3 according to the generally accepted vehicle classification in the EU. Heavy vehicles include heavy duty vehicles, touring vehicles, buses with three or more axles (M2 and N2 with trailer, M3 and N3). Powered two-wheelers are subdivided into mopeds, tricycles or quads ≤ 50 cc corresponding to categories L1, L2, L6 and into motorcycles, tricycles or quads > 50 cc designated by categories L3, L4, L5, L7. In addition to the already mentioned 4 categories of vehicles, the 5th category is also indicated – the Open category, where electric vehicles potentially fall. However, according to the Estonian CNOSSOS-EU Calculation Method Guidance Material [6], electric vehicles will not be counted until their presence on the roads is 5% of all registered vehicles. As of December 12, 2023, 746,644 vehicles were registered in Estonia, of which only 5781 have an electric motor, which is 0.77% of the total number of vehicles [29].

Rail noise refers to noise specific to transport moving on iron rails, which is relevant for both trains and trams. Estonia uses (as of March 2020) 4 types of rail transport [6]:

- electric train Stadler FLIRT;
- diesel passenger train Stadler FLIRT;
- freight train with locomotive(s);
- diesel passenger train (international, Go Rail).

The following types of trams are used in Estonia (Tallinn):

- trams in CAF Urbos (Spain);
- trams KT-4, KT-6 (Czech Republic);
- retro trams [6].

According to the CNOSSOS-EU methodology, it is recommended to calculate rail noise based on the number of train-cars, speed and other indicators, but not on the type of train. That require additional measurements for each specific case. Since no additional studies have been carried out in Estonia on this topic, the authors of the Estonian instructions for the CNOSSOS-EU [6] recommend using an intermediate method, where the choice of train types is made among the train types described at the "Reken- en Meetvoorschrift Railverkeerslawaai" (RMR) or Calculation and Measurement Regulations for Rail Traffic Noise [30] selection according to the Netherlands national computation method (SRM II). The train types include the following:

- passenger electric;
- diesel passenger train;

- freight train;
- tram [6].

3.4 Tallinn city label taxonomy

The taxonomy of Tallinn city presented below is a synthesis of the SINGA:PURA and SONYC urban taxonomies on the one hand and the CNOSSOS-EU noise pollution classification on the other hand. Figure 6 shows the noise taxonomy of the Tallinn city. Inheriting the idea of a 2-level system from SONYC, the taxonomy of Tallinn also has 2 class levels: coarse-level which are indicated in the image in soft-angled rectangles and fine-level which are indicated in sharp-angled rectangles. Green indicates classes that match both the SINGA:PURA and SONYC taxonomies, red indicates classes that match the SINGA:PURA taxonomy, and blue indicates new classes that are not present in either of the taxonomies mentioned above.

Since there is an airport and several heliports on the territory of Tallinn, compared to SINGA:PURA and SONYC taxonomies, coarse-level Aircraft has been added with the related fine-level classes "aircraft" and "helicopter". A new coarse-level "Road" has been created, which combines the "car crash" of the fine-level class, which corresponds to a similar class in coarse taxonomies in the coarse-level "Non-machinery impact" and the entire fine-level set from "Signals Alert". Potentially, all the components of the coarse-level class "Engine" could also be attributed to this class. However, the gradation of the motor by its size without specifying specific parameters, such as volume, would make this incorrect and incomparable with the classification of the CNOSSOS-EU directive. Moreover, within the taxonomy of SINGA:PURA and SONYC, "Engine" includes vehicles, but also any object that works on the basis of an engine, be it an air conditioner or an airplane. When this factor is taken into account, comparison with CNOSSOS-EU standards becomes even more incorrect.

Coarse-level "Construction" combines the previously presented categories of classes "Powered saw" and "Machinery impact". Since the sound of the "reverse beeper" class (equal to the class under "Alert signal" according to the SONYC classification) in Tallinn is more typical for trucks and heavy construction equipment, it was also included in the Coarse-level "Construction". Because SINGA:PURA and SONYC taxonomies do not describe the difference between classes 4-2 “Small/medium rotating saw” and 4-3 “Large rotating saw”, they have been merged into fine-level class 3-6 “Rotating saw”. It is also important to note that sounds related to "Construction" can be found in the area of roads if there is repair work as well as the class "heavy vehicle", although it belongs to the coarse-level "Road" which can also be found on construction sites. This is not a mistake since there are no strict distinctions between noise classes in terms of where they can be located. In this case it will contribute to a more accurate analysis of noise pollution based on real world indicators.

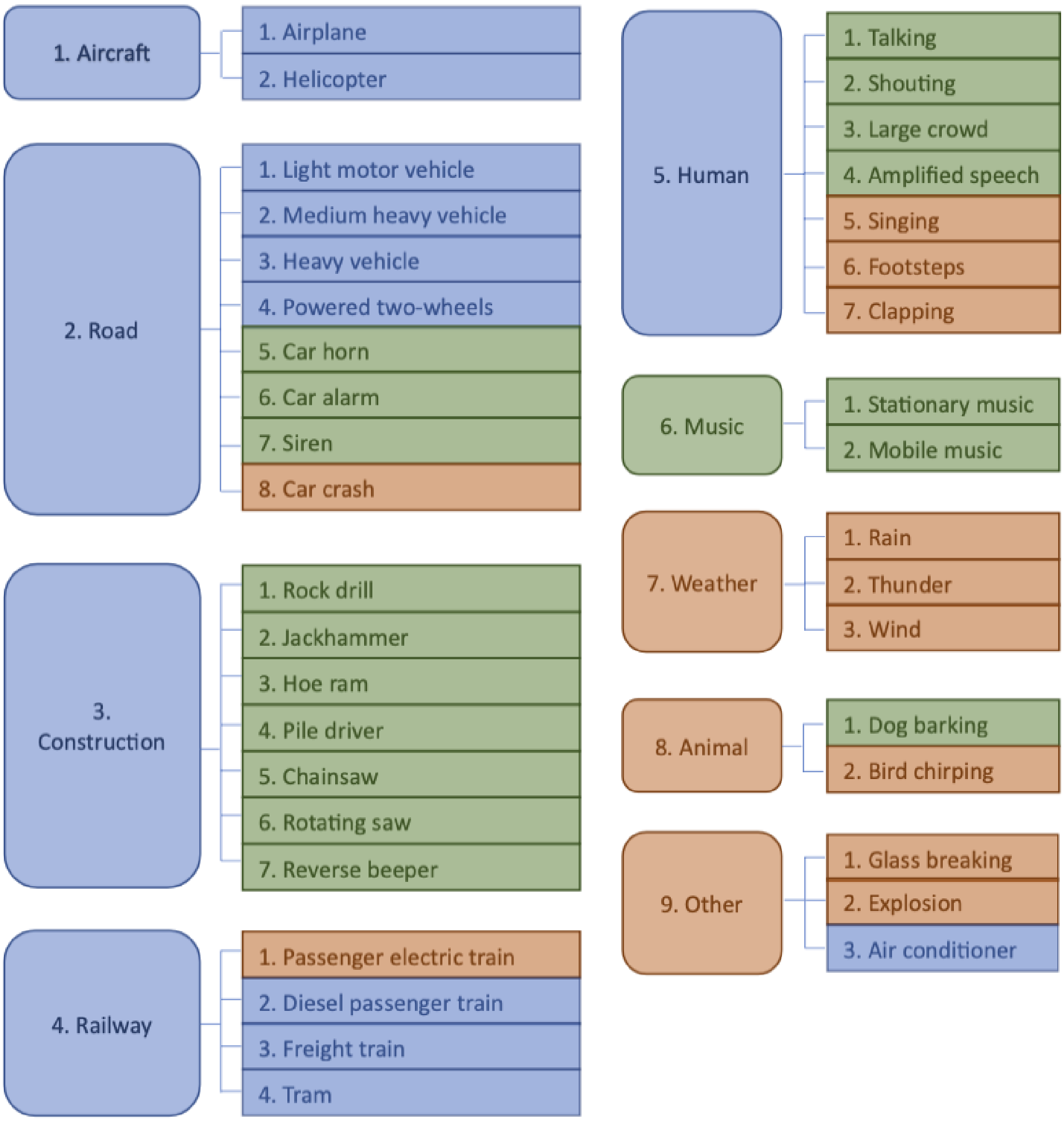

Figure 6. Tallinn city urban noise events taxonomy.

The first level class "Railway" was created, which includes "Passenger electric train" (corresponds to class 13-1 "Electric train" according to SINGA:PURA classification). In addition, according to the recommendations, "Diesel Passenger Train", "Freight Train" and "Tram" were added. This noise category should include railway vehicles described in section 3.3 applicable to Estonia.

It was decided to unite the categories "Human voice" and "Human movements" into one category "Man". The number and content of the fine-levels did not change.

Also, the coarse-level classes "Music" and "Weather" have not changed. In the coarse-level "animal" class, the "chirping insect" class was not included due to the climate and geographical location. Large loud insects that could create noise pollution are not characteristic of the Estonian region.

Classes such as "Water" with fine-level "hose pump" and "Brake" were not included in the taxonomy of Tallinn. in the first case, water pumps are little used in urban space. In the second case, the author sees an overcomplication of the taxonomy by isolating the sound of a braking vehicle into a separate category given that there is already a special category for the "Road". At the same time the creaky sound usually associated with brakes is often relevant for light motor vehicles while buses and trucks brakes sound in a different way. In addition, from an annotation point of view, it can be difficult to separate the sound event "Medium heavy vehicles" and the sound event "brake" because the braking process of a vehicle can be combined with its acceleration. Thus, the author believes that the presence of a separate one general fine-level class “Brake” is not entirely correct and is an additional taxonomic overcomplication. Instead, the author considers the sound of vehicles braking to be part of the vehicle classes themselves, such as "Light motor vehicles", "Medium heavy vehicles", "Heavy vehicles" and "Powered two-wheelers" along with acceleration sounds, which also differ for each of these classes.

In addition, the "Other" category has been revised. The classes "Screeching", "Plastic crinkling", "Cleaning", "Gear" present in SINGA:PURA were not added to the taxonomy of this work. Their interpretation can vary due to lack of the class description. As an example, the "Cleaning" class of the SINGA:PURA dataset introduces the sound of street cleaning like sweeping sidewalks or other manual street cleaning. This does not include the use of a loud mechanical devices and as a result the contribution of such activity to overall noise pollution is negligible. Instead, the fine-level classes "breaking glass" and "explosion" were assigned to the "Other" category, which in the taxonomy of

SINGA:PURA corresponding to classes 3-1 and 3-3, respectively. Also added a fine-level class "Air conditioner" – the noise from ventilation audible on the streets. That noise would be typical for restaurants, shopping, and business areas, wellness centres and hotels.

4 Dataset of Tallinn Urban Noise

This section describes the different steps in creating a dataset of urban noise of Tallinn which name is Tallinn Urban Noise dataset. This chapter describes the process of data collection and processing. The main definitions used for sound analysis are also disclosed. Finally, section describes the validation style of the created dataset and the results.

4.1 Data collection process

Once the classification structure of the dataset has been outlined, it is necessary to define, collect and process the appropriate audio data. Below is a description of the data collection process for the Tallinn Urban Noise Dataset.

4.1.1 Processing existing datasets

For processing and further use, the author selected datasets from those mentioned in the previous section where the annotation is considered as strong: UrbanSound8k and SINGA:PURA. Although the Tallinn city urban noise events taxonomy has been compiled considering the experience of already existing taxonomic practices, the names of some classes and their order is different. For example, class number "5-1" – "Car horn" in SINGA:PURA corresponds to class number "2-5" in Tallinn city urban noise events taxonomy. Table 3 shows the correspondence of classifications between UrbanSound8k, SINGA:PURA and Tallinn city urban noise events taxonomy, according to which datasets were distributed from existing datasets.

Table 3. Fine-level class correspondence between UrbanSound8k, SINGA:PURA and Tallinn city urban noise events taxonomy

Tallinn city urban sound events taxonomy class name	Tallinn city urban sound events taxonomy class number	SINGA:PURA taxonomy class number	UrbanSound8K class name
Car horn	2-5	5-1	Car horn
Car alarm	2-6	5-2	–
Siren	2-7	5-3	Siren
Car crash	2-8	3-2	–
Rock drill	3-1	2-1	Drilling
Jackhammer	3-2	2-2	Jackhammer
Hoe ram	3-3	2-3	–

Tallinn city urban sound events taxonomy class name	**Tallinn city urban sound events taxonomy class number**	**SINGA:PURA taxonomy class number**	**UrbanSound8K class name**
Pile driver	3-4	2-4	–
Chainsaw	4-5	4-1	–
Rotating saw	3-6	4-2, 4-3	–
Reverse beeper	3-7	5-4	–
Passenger electric train	4-1	13-1	–
People talking	5-1	7-1	–
People shouting	5-2	7-2	–
Large crowd	5-3	7-3	–
Amplified speech	5-4	7-4	–
Singing	5-5	7-5	–
Footsteps	5-6	8-1	–
Clapping	5-7	8-2	–
Stationary music	6-1	6-1	Street music
Mobile music	6-2	6-2	–
Rain	7-1	11-1	–
Thunder	7-2	11-2	–
Wind	7-3	11-3	–
Dog barking	8-1	9-1	Dog barking
Bird chirping	8-2	9-2	–
Glass breaking	9-1	3-1	–
Explosion	9-2	3-3	–
Air conditioner	9-3	–	Air conditioner

The sound file is an array of consecutive signal values and can be either one-dimensional or a multi-dimensional array. The number of arrays in a file is also called the number of audio channels. The number of channels in the recorded file determines the number of microphones in the recording device that can simultaneously record audio. In other words, each microphone stores the audio value in its own array. For example, the sound sensor used in the ISC2PT contains one microphone, while the number of channels in the UrbanSound8k and SINGA:PURA datasets range from 1 to 7.

The Tallinn Urban Noise dataset will be used for a machine learning algorithm that will be further maintained to microcontrollers with a single microphone and, accordingly, receive a one-dimensional array as an input. Therefore, it makes sense to transform each multi-channel file into n single-channel files, where n equals the number of channels in the original audio file. It was decided that the length of one file at the final dataset should be equal to 1 second.

A Comma Separated Values (CSV) is a plain text file that contains data in a tabular format. In a CSV file, each line represents a row, and each field within the row is separated by a comma. The first row typically contains the column names or headers and all subsequent rows contain the entries themselves. In the case of the UrbanSound8k and SINGA:PURA datasets, CSV files are used to store primarily annotation information and additional data at the discretion of the dataset creators. For example, the CSV attached to the SINGA:PURA dataset describes such characteristics as: annotator ID, filename, event_label (in numerological format), proximity, onset (beginning of a noise event in a file), offset (end of a noise event in a file), annotator remarks.

To automate the process, the author compiled a python script that:

1. iterates through the rows of the CSV file;
2. takes the event_label and, according to the data from Table 3, checks whether there is a corresponding class in the Tallinn city urban noise events taxonomy;
3. if the class match is confirmed, then based on the beginning and end of the noise event, a new file is created indicating the class according to the Tallinn city urban noise events taxonomy;
4. if the received noise event is longer than two seconds, then it is divided by the number of complete seconds in the event and each segment is saved as a separate file. If the file is more than 1 second then it is trimmed up to 1 sec;
5. if the received noise segment consists of more than one channel, save each of the channels to a separate file;
6. creates a new CSV file with records of the new filename, its Tallinn city urban noise events taxonomy class number, duration in seconds, the original filename from which the file was created, and the name of the source dataset.

The same algorithm was used by the author to convert files from UrbanSound8k to Tallinn Urban Noise dataset, except for slicing the beginning and end of the event, since the noise event in UrbanSound8k lasts the same time as the file itself. In both cases, only events whose length is greater than or equal to one second were subjected to conversion. As a result of processing, 94,539 files with total duration of 26.3 hours were obtained from the SINGA:PURA dataset and 31,639 files with total duration of 8.8 hours from UrbanSound8k dataset.

4.1.2 Recording sessions

Comparing Figure 6 and Table 3 shows that not all Tallinn city urban noise events taxonomy classes can be represented using only SINGA:PURA and UrbanSound8k. In order to make up for the lack of coarse-level "Road" classes such as "Light motor vehicle", "Medium heavy motor vehicle", "Heavy motor vehicle" and "Powered two-wheelers" several series of recording sessions were organized in relevant locations.

The noise recording sessions took place from July 15, 2021 to September 14, 2021 to collect noise data for the "industrial noise" and "road noise" classes. The sound recording procedure was carried out using a SiLabs Mighty Gecko EFR32MG12 microcontroller and Knowles SPU0410HR5H-1 microphone, which are the same hardware as used at ISC2PT project, with a signal output to a computer. Below on Figure 7 is an example image of equipment near the recording location during one of the sessions.

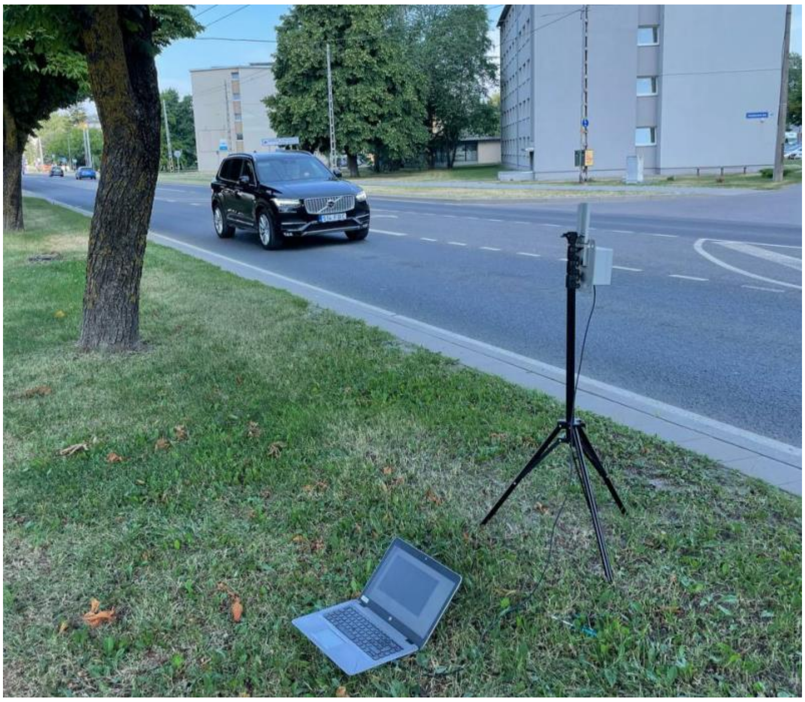

Figure 7. Sound recording process by a two-line road at Academy Street, Tallinn (15.07.2021).

The venue for the session was chosen by the principle of the favourable presence for a necessary noise and the minimum existence of noise events from undesired classes. Thus, locations were chosen near two and four lane roads to create a "road noise".

Since the sound from the real world is an analog signal but should be recorded in digital format. The Analog to Digital Converter (ADC) should be used [31], which is the peripheral of the SiLabs Mighty Gecko EFR32MG12 microcontroller and converts the analog signal into its digital value. The selected microphone contains an electric circuit that picks up the vibrations of an analog microphone's diaphragm and translates them into a voltage value in the range from 0.0V to 3.3V that is compatible with the EFR32MG12 microcontroller. This value then gets assigned a certain number within range depending on the bit-depth of a particular ADC. For example, the EFR32MG12 microcontroller has a 12-bit ADC, which means that the range of numbers is 2 to the power of 12, or from 0 to 4096 [32]. In other words, the voltage value in the range from 0.0V to 3.3V is converted to a digital value from 0 to 4095 proportionally. The number of values that the ADC can read from the microphone per second is called the sampling rate and, in this case, the 8kHz was selected to be compliant with the Tallinn Urban Noise dataset.

For recording the sound data, a ready-made Java Script program was used. The program recorded the digital values received from the ADC and has written them to a text file. A total of 3 recording series were carried out for the purpose of recording road noise. At the end of the recording, 3369 sound files each 3 second long were received, requiring further sorting and annotation.

One way to represent audio data is waveform in the time domain where the horizontal axis shows the time, and the vertical axis is the amplitude of the audio signal. Figure 8 shows an example waveform in the time domain for the sound of a passing car.

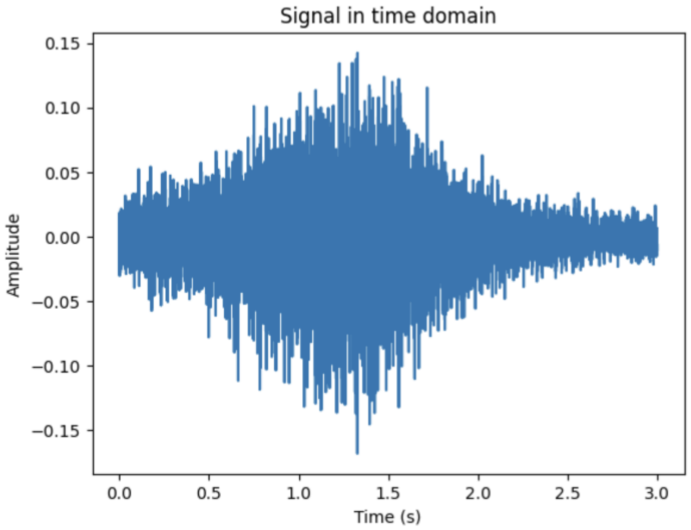

Figure 8. Sound of a passing car represented in waveform time domain.

The Discrete Fourier Transform (DFT) is used to convert the audio signal to the frequency domain [33]. Figure 9 shows a representation of sound in the frequency domain where horizontal axis denotes the range of frequencies that occur in the recording of a passing car and vertical axis indicates the magnitude or number of repetitions of a particular frequency in the recording [33].

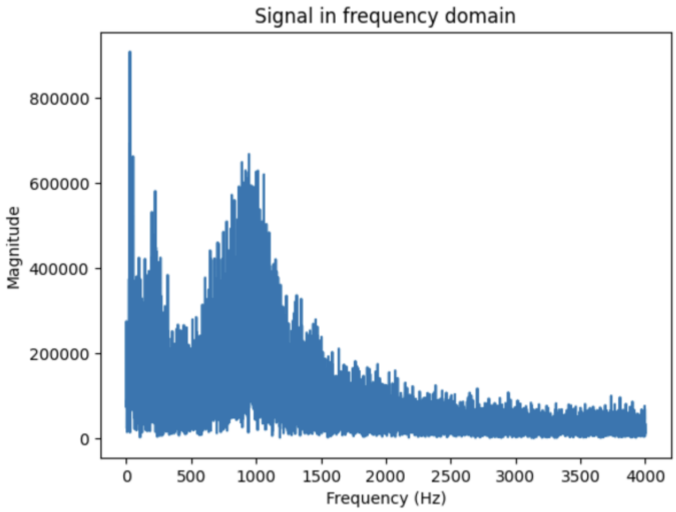

Figure 9. Sound of passing car represented in waveform in frequency domain.

To analyse how the frequency content of an audio signal changes over time, a short-time Fourier transform (STFT) is used [34]. Unlike the DFT, which provides information about the frequency content of an entire signal, the STFT analyses the frequency content of small, overlapping segments of the signal, called frames of windows [34].

The STFT is obtained by dividing the signal into short time frames, typically overlapping, and computing the DFT of each frame. Both DFT and STFT concepts are needed to understand the idea of spectrogram and its use to analyse sound. The result is a two-dimensional representation of the signal's frequency content over time, known as a spectrogram [34]. Figure 10 shows a frequency spectrogram representation of a passing car.

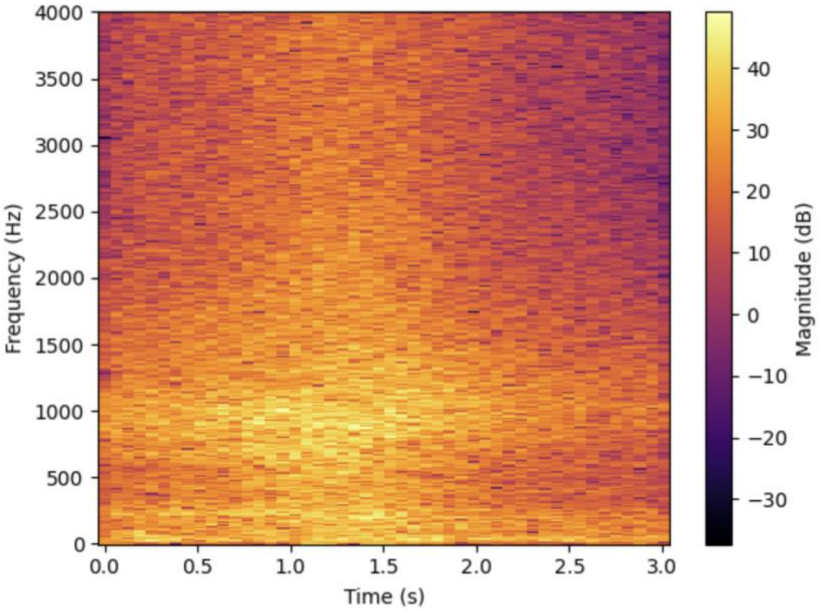

Figure 10. Sound of passing car represented in spectrogram.

The sorting of sound files obtained from road noise recording sessions was carried out by the author based on the analysis of the audio of the reproduced sound and its representation in waveform and spectrogram for a temporary domain. Recorded files were processed using the “ocenaudio” sound editor [35]. The purpose of processing was not only to allocate recorded files by class, but also the identification and elimination of extraneous sounds such as wind, birdsong, microphone interference, etc. In addition, one of the goals was to minimize the presence of files where several sound events belonging to different classes overlapping each other. Figure 11 shows the waveform and spectrogram of an audio file that contains events from two different classes. On the right side, from 0 seconds to 1.4 seconds, small strokes are visible under the number 1. This is a representation of the sound of birds singing, and in this case, the segment from the beginning of the file to 1.4 seconds was saved as a separate file and was assigned to class 8-2 "bird chirping". Event number 2 shows the sound of an approaching vehicle and in this case is also saved as a separate file.

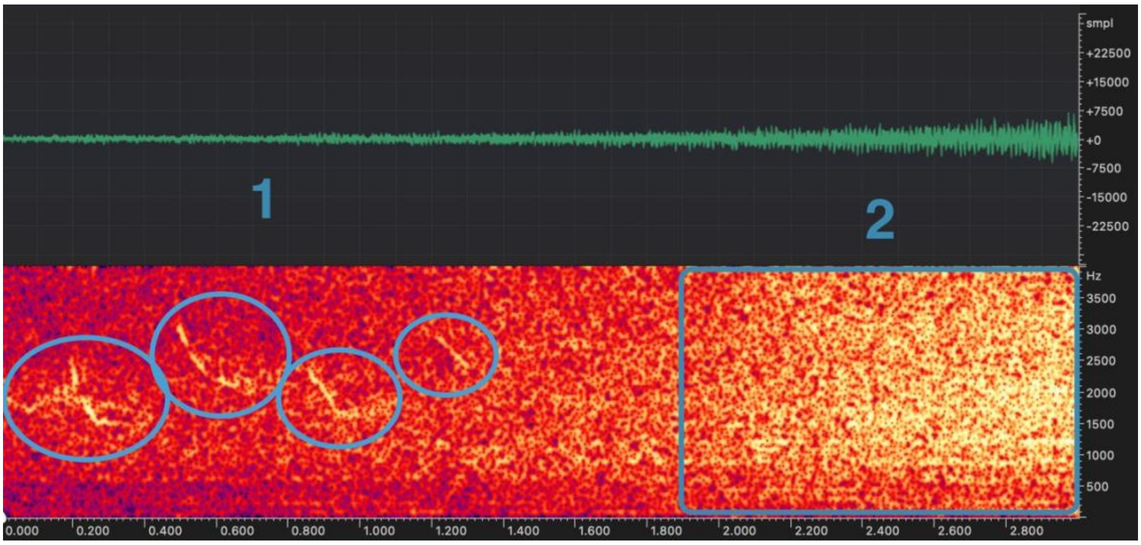

Figure 11. Waveform and spectrogram of bird chirping (1) and passing vehicle (2).

In the case of small secondary events that overlap with the dominant event in the file, the small event has been cut out. For example, in Figure 12, the main dominant event is an approaching vehicle, which starts from the first second of the file and intensifies towards the end. A minor event is highlighted in a rectangular blue frame, which in this case is a gust of wind and is well defined both on the spectrogram and in the waveform. In this case, the sound of the wind is removed using the sound editor, and only one event remains in the file – an approaching vehicle.

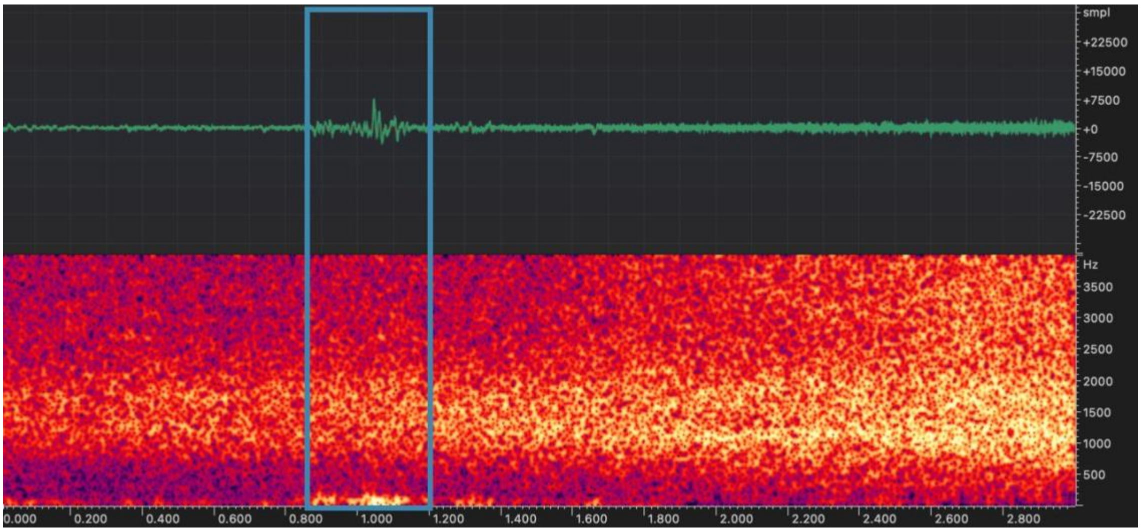

Figure 12. Waveform and spectrogram of sound where wind is blowing at the moment the car is passing by the sensor.

After the initial processing and annotation of the recorded files, the resulting files were cut into segments of 1 second. As a result of all the work, 5022 files were received. Further in the text, the name “Tallinn_recordings dataset” is used to refer to the data obtained during recording sessions. Table 4 below shows which classes are represented in the custom recorded dataset with its volume.

Table 4. Class distribution in a custom recorded dataset

Class Nr	Class name	Amount of files
1-1	Airplane	29
2-1	Light motor vehicle	3457
2-2	Medium heavy motor vehicle	733
2-3	Heavy motor vehicle	502
2-4	Powered two-wheels	21
2-5	Car horn	43
2-7	Siren	18
3-8	Reverse beeper	16
7-3	Wind	39
8-2	Bird chirping	164

As can be seen from Table 4, it was possible to collect not only "Light motor vehicle", "Medium heavy motor vehicle", "Heavy motor vehicle" and "Powered two-wheelers" but also other classes. The entry of the "Airplane" noise class can be considered especially significant since it is not represented in either SINGA:PURA or UrbanSound8k.

4.1.3 Collection from Freesound

Despite the processing of UrbanSound8k and SINGA:PURA datasets and recording sessions of some noise classes, there are still classes that are not represented in existing datasets and are difficult to collect during noise recording in the required amount. These are some subclasses of Aircraft, Road, Construction and Railway. In order to replenish these missing classes, the author searched for sound instances on the Freesound platform.

The search was performed using keywords, doing listening assessment and checking each entry for compliance. As a result of the search, 231 audio recordings with a length from 1 second to 26 minutes 41 seconds were selected for 10 subclasses. At the stage of processing the collected sound files, sound events that explicitly did not belong to the desired class were detected using the sound representation in the time domain and a spectrogram for the time domain. Then they were removed from audio files according to approach described in 4.1.2. For slicing each file into 1 second, the same approach was used as previously described in 4.1.1.

As a result of processing the collected audio files, a "Freesound collection" subset of 13,447 files was compiled. The Table 5 below shows class distribution in custom collection from Freesound.

Table 5. Class distribution in custom collection from Freesound

Class Nr	Class name	Amount of files
1-1	Airplane	1278
1-2	Helicopter	2239
2-4	Powered two-wheels	1178
2-6	Car alarm	271
3-4	Pile driver	3300
3-5	Chainsaw	583
3-6	Rotating saw	1393
4-1	Passenger electric train	878
4-3	Freight train	1633
4-4	Tram	694

4.2 Data balancing

As a result of collecting, sorting, and processing data from all four sources mentioned is this section, a dataset was obtained from 144,636 files each 1 second long and with a total duration of 40,2 hours. Figure 13 below shows the number of files that fall into each of the presented classes, where the first number refers to the coarse-level and the second number refers to the fine-level class. The figure also shows which classes from the Tallinn city urban noise events taxonomy are represented in the dataset.

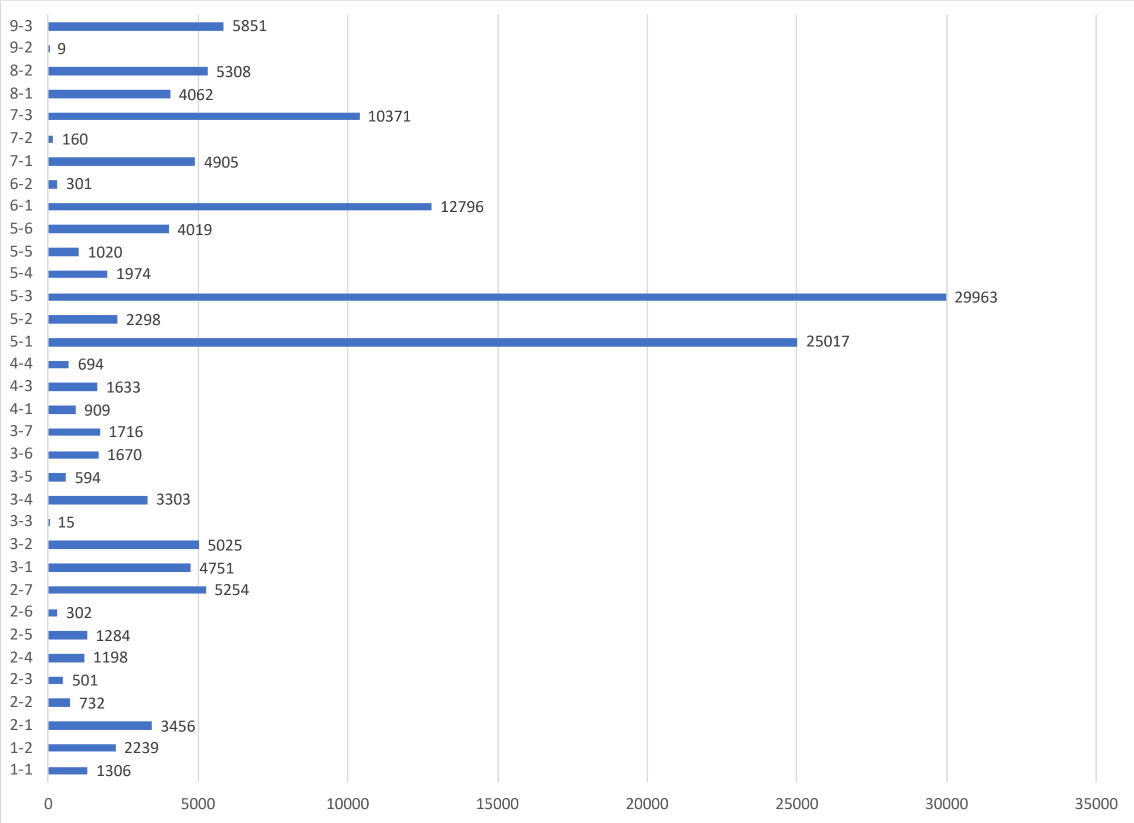

Figure 13. Distribution of files by fine-level classes.

As can be seen from Figure 13 there is a significant difference in the number of files submitted between classes 5-1 and 5-3 with the other classes. At the same time there is also a shortage for classes 3-3 and 9-2 where the number of files per class does not exceed 20. Figure 14 shows the source dataset ratio per fine-level class where one can find that a large part of the data comes from the SINGA:PURA dataset, where out of 144,636 files, the total number of SINGA:PURA files is 94,539 or 65%. UrbanSound8K accounts for a total of 31,639 files, which is 22% of the total. Custom collected dataset from Freesound platform has 13,447 files (9%) and Tallinn_recordings dataset has 5011 files (4%) in total.

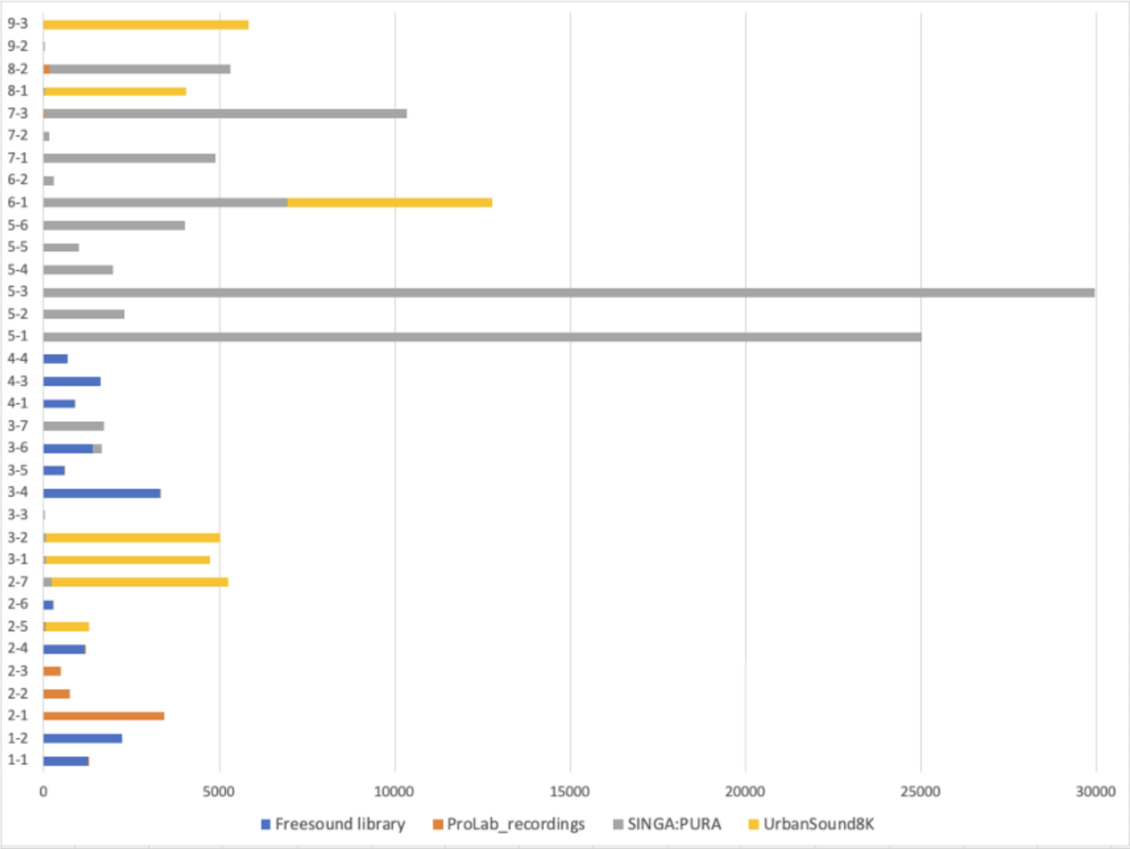

Figure 14. Source datasets ratio per fine-level class.

The dataset at Figure 14 is also characterized by heterogeneous distribution of the original data sets across classes. In most cases in one fine-level class, there is data from one Source dataset, and there are also several coarse-level classes that are compiled with data from one Source dataset. For example, the Railway class (4) consists only of the custom collected data from Freesound and the Human class (5) data consists entirely from the SINGA:PURA dataset. In the case of the classes Road (2), Construction (3), Music (6) and Animal (8), data from different sources are presented per one coarse-level class. The average number of files that falls on each coarse-level class is shown at Figure 15 where horizontal axis denotes the coarse-level classes of the class and vertical axis denotes the average number of files within one class.

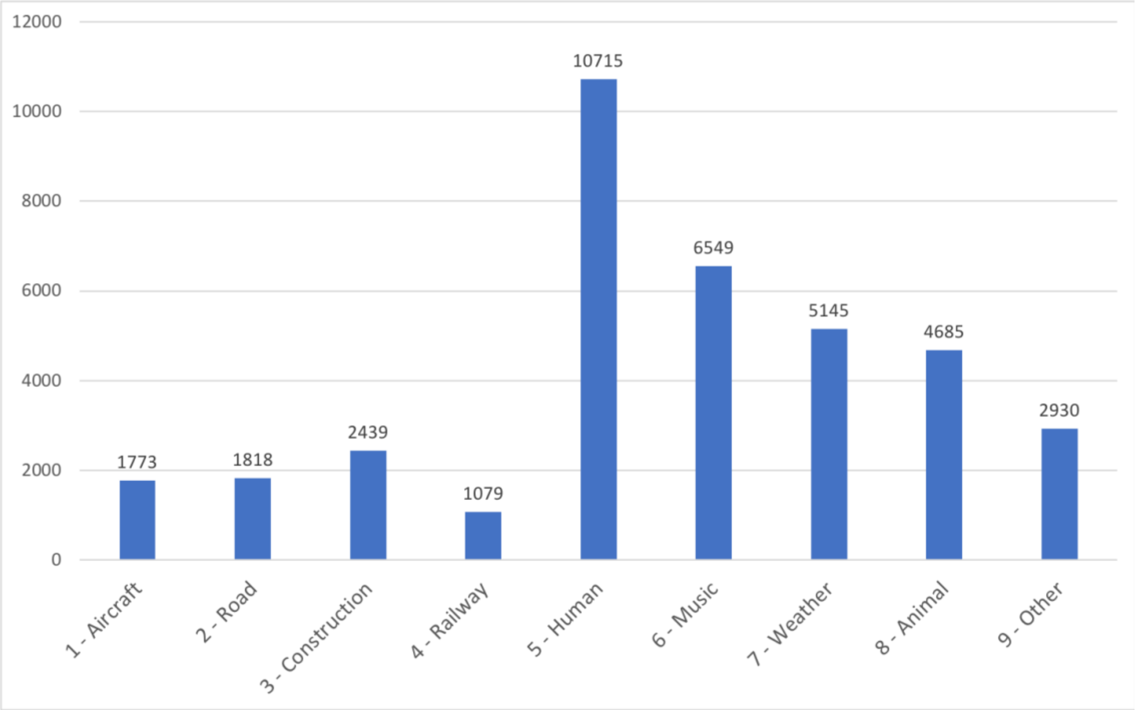

Figure 15. Average number of files inside coarse-level class.

A dataset where the number of instances for each class is approximately equal is called balanced, and vice versa, when the number of instances of one of the classes is very different from the rest, then the dataset is considered unbalanced. The use of unbalanced datasets for model training can negatively affect the accuracy of the model since it teaches the model to become biased towards one class [36].

4.2.1 Reducing the number of files

One way to reduce the significant difference in the number of files is to reduce the size of the files for those classes where the number of files exceeds the rest. Thus, the maximum number of audio instances per fine-level class has been reduced to around 6000 files and classes 3-3 and 9-2 were excluded. Figure 16 shows the data source distribution by fine-level classes after reducing the number of files after the data balancing.

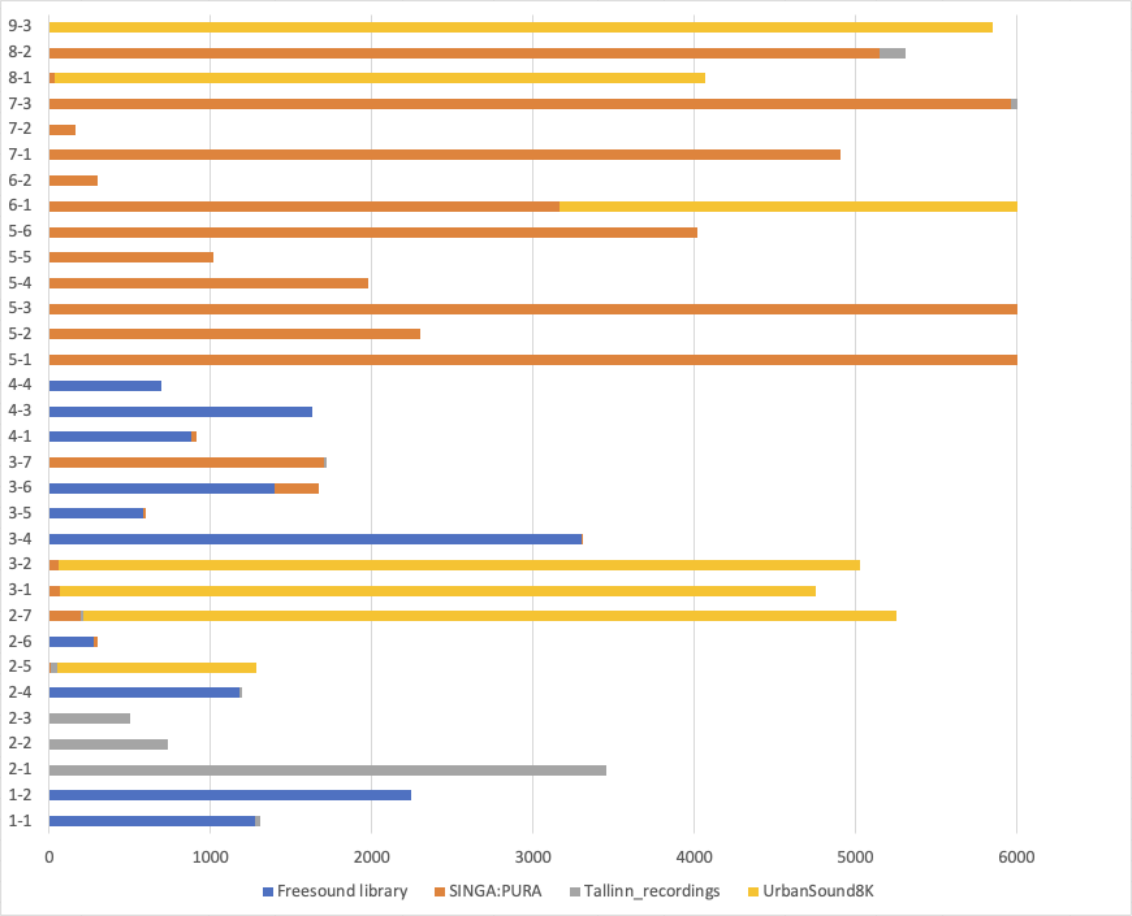

Figure 16. Data source distribution by fine-level classes.

After reducing the number of files in the Tallinn Urban Noise dataset, 90,465 files remained. The number of files from the SINGA:PURA dataset has decreased to 43,368 which is 47,9% of the total number of files. The number of UrbanSound8k files has decreased to 28,639 which is 31,6% of the total. The number of Tallinn_recordings dataset and Freesound files has not decreased and amounts to 5,5% and 14,9% respectively.

4.2.2 Data augmentation

In addition to reducing the number of files, there are also methods to fill in the missing amount of data in the dataset. Audio data augmentation is a technique used in machine learning to enhance the quantity of audio datasets for training of machine learning models. The method involves applying various transformations to the original audio signal, such as time stretching, pitch shifting, adding noise, or changing the speed and volume in a way that the semantic value of original signal is not changed [37].

To augment audio data, the author used the Python package named "tsaug" [38] which offers a set of augmentation methods for time series. Following augmentation methods developed by Arundo Analytics [39] were applied:

- “AddNoise”: adds independent and identically distributed noise to the existing audio at each point in time of the time series.
- Convolve: applies convolution operations to the input time series data. The process involves sliding a convolutional filter over the time series, creating new data points by combining the original data with the filter's effects.
- Drift: introduces random and gradual shifts in the time series values, deviating them from their initial state. The magnitude of this shift is determined by the maximum drift allowed and the number of points affected by the drift.
- Quantize: discretize the time series into predefined levels by rounding the values in the time series to the nearest level within the set.
- Pool: involves down-sampling or pooling the data by selecting specific segments or points at regular intervals not affecting the total length of time series.

For each fine-level class, the required number of augmented files was determined so that the total number of files in the class reached 6000. The "tsaug" package allows to create a function which applies above methods to the audio file with specified probability parameter. Within this function it is definable to which extend each method is applied to audio file. The one can find the Python function parameters at Figure 17 which was used during the augmentation process.

```
my_augmenter = (
      AddNoise(scale=(0.01, 0.05)) @0.5
      + Convolve(window=["boxcar", "triang",
      "blackman", "hamming", "hann", "bartlett"])
      + Quantize(n_levels=(7, 20), prob=0.5)
      + Drift(max_drift=(0.02, 0.3)) @0.5
      + Pool(size=(1, 3)) @0.5)
```

Figure 17. Function with methods for random augmentation.

As shown at Figure 17 the “AddNoise” method adds noise value in range of 0.01 to 0.05 with 50% probability rate. Then the Convolve method is applied using one of the filters listed inside the brackets. Quantize method is configured in the way to quantize the time series into levels in range of 7 to 20 with 50% probability rate. For the Drift method the maximum drift value was set in range of 0.02 to 0.3 and 50% probability rate of use. The size parameter at Pool method was used to set the size of pooling window which might be 1 or 3 and. The Pool method was applied to augmented file with the 50% probability.

Thus, “my_augmenter” function was applied to audio files at each fine-level class where the total amount of files was below 6000. The choice of parameters was tested by the author of the thesis to select optimal settings at which the sound remained close to the original value. Figure 18 shows the average number of files inside coarse-level class after augmentation was finished.

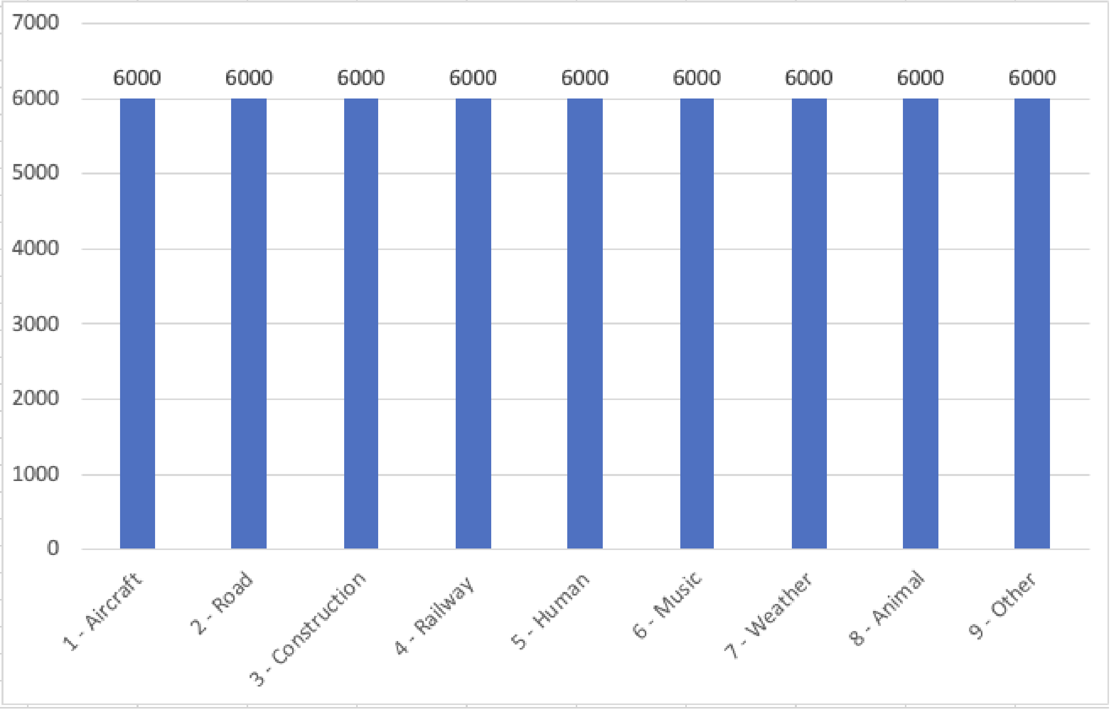

Figure 18. Average number of files inside coarse-level class after augmentation.

As a result of the augmentation process, 101,535 files from different classes were added to the dataset. A record about each new file is added to the general CSV file of the final dataset, including the source filename and note if this file was created with augmentation. Figure 19 shows the ratio of original data to augmented data in the final version of the dataset.

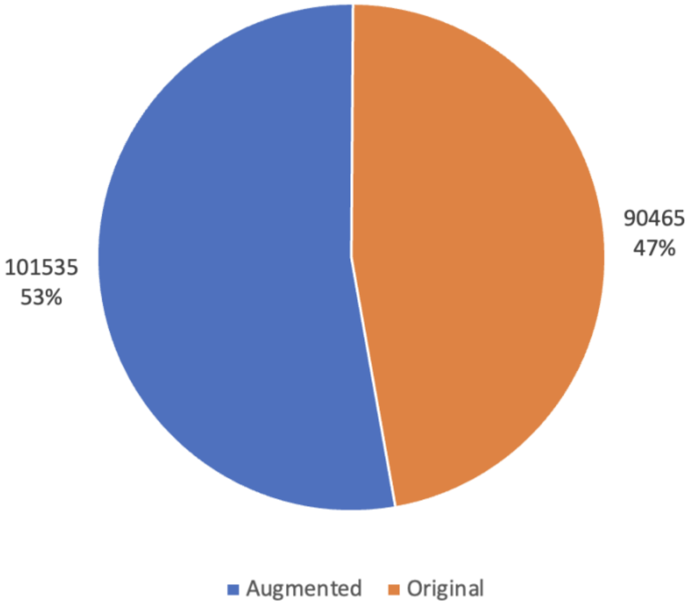

Figure 19. Augmented to original files ratio inside the Tallinn Urban Noise dataset.

As can be seen above, the amount of augmented data is 53% of the total number of files inside the dataset.

4.3 Tallinn Urban Noise Dataset validation

Tensorflow is an open-source software library for machine learning developed by Google to solve problems of building and training a neural network to automatically find and classify images, achieving the quality of human perception. It also provides a comprehensive ecosystem of tools, libraries, and community resources to help researchers and developers build and deploy machine learning applications [40]. Since Tensorflow also offers solutions for transforming ready-made machine learning models into a format suitable for running on microcontrollers, this library can potentially be used for further work within the ISC2PT project [8].

In order to validate the Tallinn Urban Noise Dataset, the training material proposed by Tensorflow for Simple audio recognition was used [41]. This tutorial shows how to pre-process WAV audio files to build and train a basic automatic speech recognition (ASR) model to recognize ten different words. For the task of validating the dataset, the author of the thesis replaced the audio dataset of 10 words proposed in the material with the Tallinn Urban Noise Dataset so that the model would recognize not words, but the noise classes described earlier. The dataset was divided into three subsets for training, validation, and evaluation of the machine learning model.

The model consists of input layer in the shape of (61, 129, 1), where 61 is the height, 129 is the width, and 1 is the number of channels, resizing layer, normalization layer, 2 convolutional layers, max pooling layer, 2 dropouts, 1 flattened layer and 2 dense layers. Below at Figure 20 the one can find the structure of used model for the dataset validation task.

```
Input shape: (61, 129, 1)
Model: "sequential_4"
_________________________________________________________________
 Layer (type)                Output Shape              Param #   
=================================================================
 resizing_2 (Resizing)       (None, 32, 32, 1)         0         
                                                                 
 normalization_2 (Normaliza  (None, 32, 32, 1)         3         
 tion)                                                           
                                                                 
 conv2d_5 (Conv2D)           (None, 30, 30, 32)        320       
                                                                 
 conv2d_6 (Conv2D)           (None, 28, 28, 64)        18496     
                                                                 
 max_pooling2d_4 (MaxPoolin  (None, 14, 14, 64)        0         
 g2D)                                                            
                                                                 
 dropout_8 (Dropout)         (None, 14, 14, 64)        0         
                                                                 
 flatten_13 (Flatten)        (None, 12544)             0         
                                                                 
 dense_34 (Dense)            (None, 128)               1605760   
                                                                 
 dropout_9 (Dropout)         (None, 128)               0         
                                                                 
 dense_35 (Dense)            (None, 9)                 1161      
                                                                 
=================================================================
Total params: 1625740 (6.20 MB)
Trainable params: 1625737 (6.20 MB)
Non-trainable params: 3 (16.00 Byte)
```

Figure 20. Model structure for the dataset validation task.

Machine learning model training involves iteratively feeding training data through the network, calculating predictions, measuring prediction errors using a defined loss function, backpropagating these errors to update model parameters through an optimization algorithm (e.g., gradient descent), and repeating this process (typically over multiple epochs) to improve the model's performance until convergence, aiming to minimize the loss function and achieve accurate predictions on data not used for training.

The number of data used in one cycle of machine learning model is named the batch which value was set to 64 audio samples per batch. The training of the machine learning model with one batch for one cycle is named an epoch. The model described at Figure 20 was trained over 21 epochs.

A loss function, also known as a cost function or an error function, is a mathematical expression that quantifies the difference between the predicted values generated by a machine learning model and the true or ground-truth values of the target variable [42]. The loss function serves as a measure of the model's performance and is used to guide the training process by providing a feedback signal for updating the model's parameters (weights and biases). However, to ensure that the model is learning to generalize and not simply memorizing the training data (i.e., overfitting), it's essential to evaluate the model's performance on data it hasn't seen before using the validation dataset. The validation loss provides a quantitative measure of how well the model is performing on this unseen validation set. It helps to assess if the model is learning the underlying patterns of the data or if it's failing to generalize [43]. The goal is to minimize this loss on the validation set, indicating that the model is achieving good performance and can make accurate predictions on new, unseen data. Below at Figure 21 one can see dynamics of loss function and loss function validation during the training.

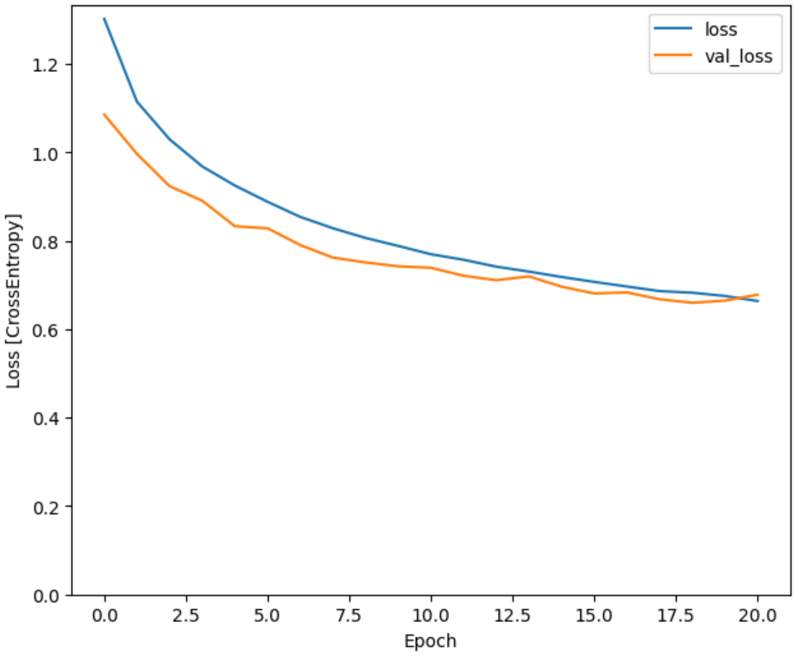

Figure 21. Loss function and loss function validation change during the training process.

The loss function value started from 1.3017 and validation of loss function initial value is 1.0854. After 21 training epoch loss function dropped to the 0.6638 and validation of loss function value changed to 0.6781. A rise in the validation loss compared to the training loss could signal overfitting, where the model starts to tailor itself too closely to the training data and performs poorly on data it hasn't been trained on. This behaviour indicates the need to stop the training process [43].

Model accuracy measures the proportion of correctly predicted instances out of the total predictions made by the model. It is a percentage representing the ratio of correct predictions to the total predictions. Validation accuracy specifically gauges the accuracy of the model based on the validation dataset offering insights into how well the model generalizes to unseen data [44]. Below the Figure 22 shows the dynamics of model accuracy and validation accuracy during the training process.

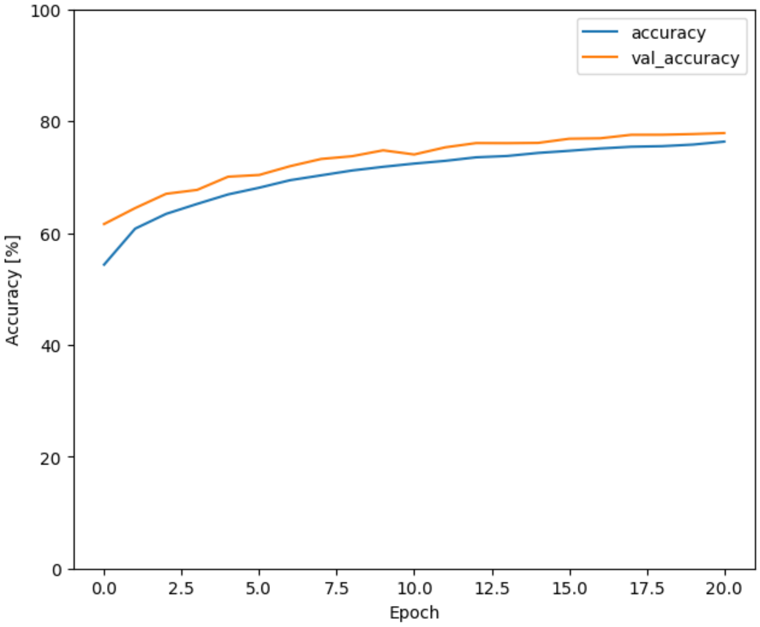

Figure 22. Model accuracy and validation accuracy change during the training process.

Within the training process model accuracy changed from 54.34% to 76.34% and validation accuracy from 61.6% to 77.87%. The trained model was evaluated with 0.673 loss and 78.1% accuracy.

Confusion matrix is a common tool for visual analysis of the machine learning model performance in the context of binary or multiclass classification problems. It is a two-dimensional matrix where the rows indicate the instances in an actual class and the columns indicate the instances in a predicted class and is formed based on the test set of the data for which the true values are known. The clearer the diagonal alignment between actual and predicted classes, the higher the model's performance is regarded [44]. Below at Figure 23 one can find the confusion matrix showing the model's classification performance for each coarse-level class in the test set.

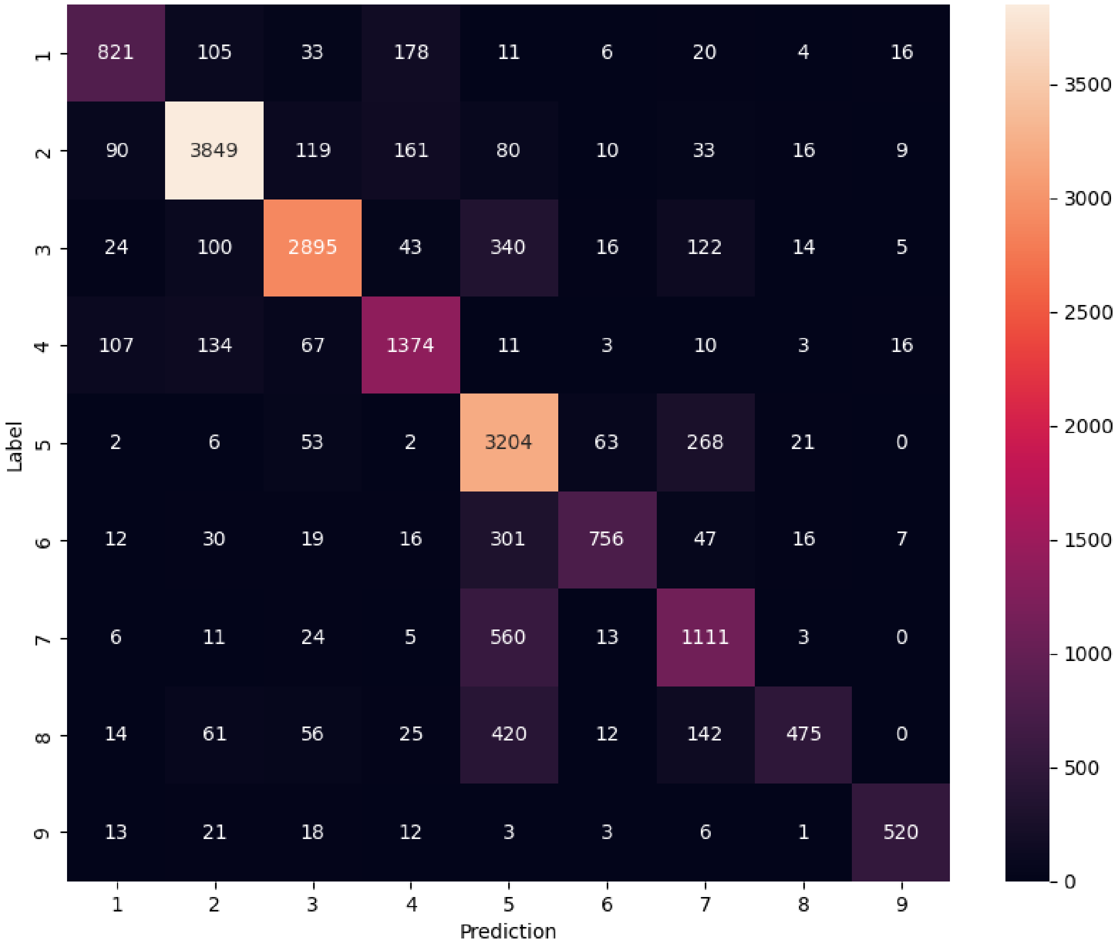

Figure 23. Confusion matrix of machine learning model based on Tallinn Urban Noise dataset.

From the confusion matrix at Figure 23, it can be seen the overall distribution of the predicted classes generally coincides with the real labels. Nevertheless, it is clear that coarse-level class 5 (Human) is in some cases also predicted as coarse-level classes 7 (Weather) and 8 (Animal) and a little less often as coarse-level classes 3 (Construction) and 6 (Music). Also, in the opposite direction, false predictions of coarse-level class 7 as 5 and to a lesser extent as 8 and 3 are observed. This may indicate insufficient quality of data for classes 5 and 7 in general. Looking at the Figure 16 one can see that both classes are formed entirely on the basis of the SINGA:PURA dataset which, although strongly-labelled, is also polyphonic. This indicates that sound events from different classes can overlap. Such aspect might be a potential reason the model predicts coarse-level class 7 and 5 with less accuracy. At the same time, for the classes 2 (Road), 3, 6 and 8 the SINGA:PURA is no longer dominant dataset and is combined with data from other monophonic datasets. According to the confusion matrix, the model predicts these classes quite accurately and the results are comparable to classes where the data is presented only from monophonic datasets.

Thus, one can assume that the mere presence of polyphonic data for a class does not worsen the model's performance if the class also has a monophonic data in sufficient quantities. At the same time, comparing the results of training a model based on monophonic audio and polyphonic data, the author is assuming that, within the scope of the presented dataset, it is more effective to use monophonic data or a combination of them. Therefore, to improve the quality of the data, the author proposes, as part of further work, to supplement classes 5 and 7 with data that are monophonic.

In order to verify the model's prediction output, the 1 sec audio WAV file of passing car (that is not the part of Tallinn Urban Noise dataset) was used as an input of. The model's output is a one-dimensional matrix with the shape of (1, 9) where 9 corresponds to the number of urban noise classes. Each number shows how model evaluates the probability of noise class to be presented within the provided audio sample in Float data type. In order to provide more understandable result, the Softmax function was applied. It ensures that the sum of the probabilities across all classes is equal to 1 and thereby transforming the values into the range from 0 to 1. Below in Figure 24 one can find the model output values after Softmax function was applied.

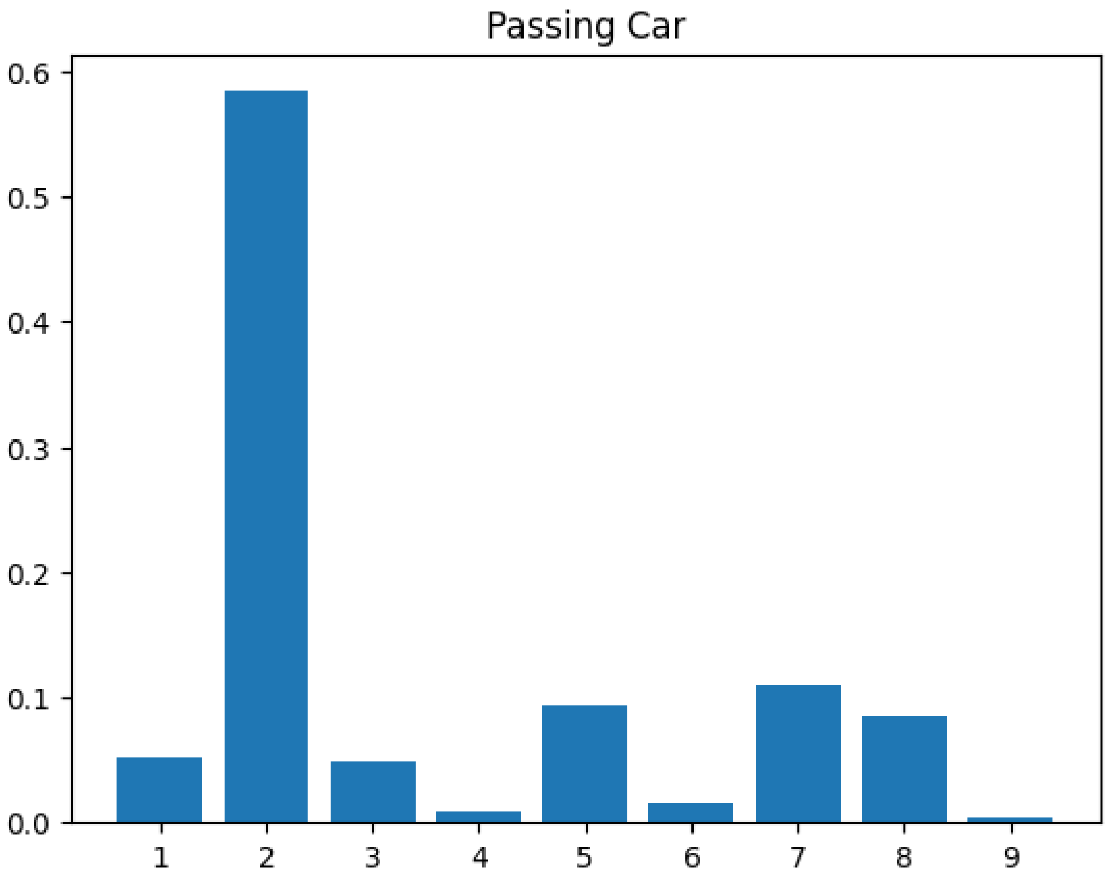

Figure 24. Model output with Softmax function applied. Input: 1 sec audio WAV file of passing car.

Considering Figure 24 the model predicted that it is more likely that the provided audio contained a second coarse-level class “Road”. Thus, the model’s prediction considered with reality and the test was successful.

4.4 Tallinn Urban Noise Dataset description and analysis

The Tallinn Urban Noise Dataset consists of 192,000 WAV files each 1 second long. The total duration of the dataset is 53.33 hours. Each audio file has an audio frequency of 8000 Hz and 16 bits per sample, which corresponds to the audio characteristics of the Knowles SPU0410HR5H-1 microphone used at ISC2PT project [8].

The structure of the dataset is a system of two levels of classes of noise pollution in the urban space of the city of Tallinn based on Tallinn city urban noise events taxonomy at Figure 6. This classification system takes into account both the results of previous studies and the noise classification criteria approved by the European Union. This strategy allowed pre-existing urban noise audio datasets to be combined with EU regulations.

The Tallinn Urban Noise Dataset contains 32 fine-level classes where the number of files per class is 6000. Although all coarse-level classes are represented in the dataset, some fine-level classes are not represented due to lack of data, which requires further refinement in subsequent work. Thus, classes 3-3 “Hoe ram”, 4-2 “Diesel passenger train”, 5-7 “Clapping” corresponding to coarse-level class “Human”, 9-1 “Glass breaking” and 9-2 “Explosion” are not present in the dataset.

The dataset is accompanied by a file in CSV format that contains records about each file in the dataset. The document contains information such as file name, class number, duration in seconds, source filename, source dataset and whether the file was augmented (True / False). In the case of Freesound resource, the source filename column contains a URL link to the source file. The dataset along with the CSV file can be found in the GitHub repository [45]. The unbalanced version of the dataset corresponding to the Figure 13 can be found at the same git repository along with the CSV file.

The audio data is a combination of pure audio files without additional audio processing and augmented data in a ratio of 47% to 53%, respectively. This thesis does not answer the question of whether such a data ratio has a positive or negative impact on the accuracy of the machine learning model. However, since the CSV file contains data about each of the files in the dataset, one can adjust the ratio of augmented data to real data based on their needs and goals. Thus, Tallinn Urban Noise Dataset can be employed for further investigation regarding the impact of the quantity of augmented data within a dataset on the machine learning model's accuracy.

The dataset consists of data collected using:

- processing of existing urban noise datasets SINGA:PURA and UrbanSound8k,
- recording sessions (marked in the CSV file as “ProLab recordings”)
- manual data selection from Freesound.

The data from the SINGA:PURA dataset is considered polyphonic, and all other sources are monophonic. It can be assumed that this speciality influenced the validation results of the Tallinn Urban Noise Dataset illustrated at Figure 22, where classes entirely or mostly from the SINGA:PURA data are recognized worser than the others. This assumption requires additional research and may become the basis for improving data quality within the future works.

5 Summary

This thesis aimed to develop and validate an audio dataset that can be used for the urban noise source recognition task in the Tallinn city. The research problem was to outline the taxonomy system of urban noise specific to the Tallinn city. Another research problem was to study such methods of collecting real world audio data of urban noise as processing existing datasets (UrbanSound8k and SINGA:PURA), data search through the open platform Freesound and noise recording sessions.

The research provides an overview on existing datasets containing urban noise and how audio data is collected. Additionally, it presents a review of taxonomy practices used in the industry. One of the results of the study is the two-level Tallinn city urban noise events taxonomy, which combines the practices of SINGA:PURA and SONYC urban taxonomies as well as the CNOSSOS-EU noise pollution classification. Thus, datasets based on this taxonomy can both inherit part of already existing datasets and comply with the Environmental Noise Directive (2002/49/EC) standards adopted in the European Union.

As another result, the study offers a Tallinn urban noise dataset consisting of 192,000 files with a total duration of 53.33 hours. The audio data comprises unaltered audio files alongside augmented data, with a distribution ratio of 47% or 25 hours for the former and 53% or 28.28 hours for the latter. The dataset validation procedure included training a small machine learning model for audio recognition. During the training process, the model accuracy changed from 54.34% to 76.34%, and the verification accuracy – from 61.6% to 77.87%. The trained model was evaluated with a loss of 0.673 and an accuracy of 78.1%.

Based on the results of the confusion matrix, the author suggests that this result can be improved by diluting the data from classes 5 and 7 with data that will be characterized as monophonic. Moreover, the proportion of original data in relation to augmented data might be served as a foundation for exploring how the quantity of augmented data in a dataset affects the accuracy of the machine learning model.

Bibliography

[1] Centre for Environment & Health (BON), Living & Working Environments (LWE), "ENVIRONMENTAL NOISE GUIDELINES for the European Region," World Health Organization. Regional Office for Europe, Copenhagen, 2018.

[2] The European Parliament and The Council of The European Union, "DIRECTIVE 2002/49/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 25 June 2002 relating to the assessment and management of environmental noise," Official Journal of the European Communities, 2002.

[3] S. Kephalopoulos, M. Paviotti and F. Anfosso-Lédée, Common Noise Assessment Methods in Europe (CNOSSOS-EU), Luxembourg: Publications Office of the European Union, 2012, p. 180.

[4] Estonian Land Department, "Estonia Noise Map," [Online]. Available: https://xgis.maaamet.ee/xgis2/page/app/myrakaart. [Accessed 15 March 2023].

[5] Estonian Land Department, "Müraandmete kaardirakenduse kirjeldus," 05 April 2023. [Online]. Available: https://geoportaal.maaamet.ee/est/Kaardirakendused/Murakaart/Muraandmete-kaardirakenduse-kirjeldus-p587.html. [Accessed 20 April 2023].

[6] M. Ründva, Juhend: Strateegilised mürakaardid. CNOSSOS-EU arvutusmeetodi juhendmaterjal, Keskonnaõiguse keskus, 2020, p. 40.

[7] "Smart Environment Networking Technologies," [Online]. Available: https://www.etis.ee/Portal/Projects/Display/524915ce-7aa5-43a2-9463-7d519689ef5b?lang=ENG. [Accessed 16 November 2022].

[8] "Intelligent Smart City and Critical Infrastructure Protection Technologies ISC2PT II," [Online]. Available: https://www.etis.ee/Portal/Projects/Display/96569b7f-a42a-422f-bd8f-97cbb3a483b1?lang=ENG. [Accessed 16 November 2022].

[9] J. Kaugerand, “Mediated Interactions for Collection and Exchange of Situational Information in Smart Environments,” Tallinn University of Technology, Tallinn, 2020.

[10] J. Salamon, D. MacConnell, M. Cartwright, P. Li and J. P. Bello, “Scaper: A library for soundscape synthesis and augmentation,” in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2017.

[11] R. Scheibler, E. Bezzam and I. Dokmanić, “Pyroomacoustics: A Python package for audio room simulations and array processing algorithms,” *arXiv*, 2017.

[12] K. Ooi, K. N. Watcharasupat, S. Peksi, F. A. Karnapi, Z.-T. Ong, D. Chua, H.-W. Leow, L.-L. Kwok, X.-L. Ng, Z.-A. Loh and W.-S. Gan, “A Strongly-Labelled Polyphonic Dataset of Urban Sounds with Spatiotemporal Context,” in *Proceedings of the 2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2021.

[13] X. Serra, F. Font, J. Pons, X. Favory and E. Fonseca, “FSD50K: An Open Dataset of Human-Labeled Sound Events,” *EEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 829-852, 2022.

[14] J. Salamon, C. Jacoby and J. P. Bello, “A Dataset and Taxonomy for Urban Sound Research,” in *22nd ACM International Conference on Multimedia*, Orlando, 2014.

[15] J. Salamon, J. Bello, C. Silva, O. Nov, R. DuBois, A. Arora, C. Mydlarz and H. Doraiswamy, “SONYC: A System for the Monitoring, Analysis and Mitigation of Urban Noise Pollution,” *Communications of the ACM*, 19 May 2018.

[16] K. Ooi, K. N. Watcharasupat, S. Peksi, Z.-T. Ong, D. Chua, H.-W. Leow, L.-L. Kwok, X.-L. Ng, Z.-A. Loh and W.-S. Gan, “SINGA:PURA (SINGApore: Polyphonic URban Audio),” 04 September 2021. [Online]. Available: https://doi.org/10.21979/N9/Y8UQ6F. [Accessed 25 March 2023].

[17] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal and M. Ritter, “Audio Set: An ontology and human-labeled dataset for audio events,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017.

[18] M. Cartwright, A. E. M. Mendez, A. Cramer, V. Lostanlen, G. Dove, H.-H. Wu, J. Salamon, O. Nov and J. Bello, “SONYC Urban Sound Tagging (SONYC-UST): A Multilabel Dataset from an Urban Acoustic Sensor Network,” in *Detection and*

Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019), New York, 2019.

[19] M. Cartwright, J. Cramer, A. Mendez, Y. Wang, H.-H. Wu, V. Lostanlen, M. Fuentes, G. Dove, C. Mydlarz, J. Salamon, O. Nov and J. Bello, “SONYC-UST-V2: An Urban Sound Tagging Dataset with Spatiotemporal Context,” in *Detection and Classification of Acoustic Scenes and Events 2020*, Japan, 2020.

[20] F. Font, G. Roma and X. Serra, “Freesound Technical Demo,” in *MM'13. Proceedings of the 21st ACM international conference on Multimedia*, Barcelona, 2013.

[21] E. Fonseca, J. Pons, X. Favory, F. Font, D. Bogdanov, A. Ferraro, S. Oramas, A. Porter and X. Serra, “Freesound Datasets: A Platform for the Creation of Open Audio Datasets.,” in *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR*, Suzhou, 2017.

[22] R. Serizel and N. Turpault, “Sound Event Detection from Partially Annotated Data: Trends and Challenges,” in *IcETRAN conference*, Srebrno Jezero, Serbia , 2019.

[23] S. Perry, V. Tiwari, N. Balaji, E. Joun, J. Ayers, M. Tobler, I. Ingram, R. Kastner and C. Schurgers, “Pyrenote: a Web-based, Manual Annotation Tool for Passive Acoustic Monitoring,” in *2021 IEEE 18th International Conference on Mobile Ad Hoc and Smart Systems (MASS)*, Denver, CO, USA, 2021.

[24] S. Adavanne, H. Fayek and V. Tourbabin, “Sound Event Classification and Detection with Weakly Labeled Data,” in *Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE 2019)*, New York, 2019.

[25] N. Turpault, R. Serizel and E. Vincent, “Limitations of Weak Labels for Embedding and Tagging,” in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* 2020.

[26] “Freesound Annotator,” Universitat Pompeu Fabra Barcelona, Music Technology Group (MTG), Freesound, [Online]. Available: https://annotator.freesound.org. [Accessed 20 March 2023].

[27] MathWorks, “yamnetGraph. Graph of YAMNet AudioSet ontology,” MathWorks, [Online]. Available: https://ch.mathworks.com/help/audio/ref/yamnetgraph.html. [Accessed April 2023].

[28] D. Ellis and P. Brossier, “Audio Set Ontology,” 8 March 2017. [Online]. Available: https://github.com/audioset/ontology. [Accessed 18 March 2023].

[29] Transpordiamet, “Sõidukite statistika,” [Online]. Available: https://www.transpordiamet.ee/soidukite-statistika. [Accessed 12 December 2023].

[30] Wölfel Meßsysteme, Software GmbH & Co, “AR-INTERIM-CM. Calculation and measurement guidelines for rail transport noise,” 1996. [Online]. Available: https://www.schiu.com/utilidades/artigos/MInterino-SRMII.pdf. [Accessed 08 April 2023].

[31] S. W. Smith, “CHAPTER 3 - ADC and DAC,” in *Digital Signal Processing,* Boston, Newnes, 2003, pp. 35-37.

[32] Silicon Laboratories, “EFR32MG12 Based Modules (Series 1),” [Online]. Available: https://www.silabs.com/wireless/zigbee/efr32mg12-series-1-modules#. [Accessed 30 April 2023].

[33] S. W. Smith, “CHAPTER 8 - The Discrete Fourier Transform,” in *Digital Signal Processing,* Boston, Newnes, 2003, pp. 141-168.

[34] M. Ashouri, F. Faria da Silva and C. Bak, “Application of Short-Time Fourier Transform for Harmonic-Based protection of Meshed VSC-MTDC Grids,” *The Journal of Engineering,* vol. 16, pp. 1439-1440, 2018.

[35] “Ocenaudio,” [Online]. Available: https://www.ocenaudio.com/whatis. [Accessed 20 April 2023].

[36] I. Pratama, Y. Pristyanto and P. T. Prasetyaningrum, “Imbalanced Class handling and Classification on Educational Dataset,” in *4th International Conference on Information and Communications Technology (ICOIACT)*, 2021.

[37] L. Ferreira-Paiva, E. Alfaro-Espinoza, M. A. Vinicius, B. F. Leonardo and V. A. N. Rodolpho, “A survey of data augmentation for audio classification.,” XXIV Brazilian Congress of Automatics (CBA), 2022.

[38] “tsaug,” Arundo, 2020. [Online]. Available: https://tsaug.readthedocs.io/en/stable/#. [Accessed 27 April 2023].

[39] Arundo Analytics, “Tsaug library. References,” [Online]. Available: https://tsaug.readthedocs.io/en/stable/references.html. [Accessed 21 October 2023].

[40] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, R. Jozefowicz and J, *TensorFlow: Large-scale machine learning on heterogeneous systems,* Zenodo, 2015.

[41] TensorFlow, "Simple audio recognition: Recognizing keywords," [Online]. Available: https://www.tensorflow.org/tutorials/audio/simple_audio. [Accessed 14 October 2023].

[42] A. Agrawal, "Loss Functions and Optimization Algorithms. Demystified.," 29 September 2017. [Online]. Available: https://medium.com/data-science-group-iitr/loss-functions-and-optimization-algorithms-demystified-bb92daff331c. [Accessed September 2023].

[43] Baeldung, "Training and Validation Loss in Deep Learning," 15 September 2023. [Online]. Available: https://www.baeldung.com/cs/training-validation-loss-deep-learning#:~:text=On%20the%20contrary%2C%20validation%20loss,the%20performance%20of%20the%20model.. [Accessed 30 September 2023].

[44] A. F. Gad, "Evaluating Deep Learning Models: The Confusion Matrix, Accuracy, Precision, and Recall," 2020. [Online]. Available: https://blog.paperspace.com/deep-learning-metrics-precision-recall-accuracy/. [Accessed 30 September 2023].

[45] J. Kärner, "Tallinn Urban Noise," September 2023. [Online]. Available: https://github.com/jelikaer/TallinnUrbanNoise/tree/main. [Accessed 20 October 2023].

[46] A. Shah, A. Kumar, A. G. Hauptmann and B. Raj, *A Closer Look at Weak Label Learning for Audio Events,* arXiv:1804.09288, 2018.

[47] E. Fonseca, M. Plakal, D. P. W. Ellis, F. Font, X. Favory and X. Serra, "Learning Sound Event Classifiers from Web Audio with Noisy Labels," in *CASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* 2019.

[48] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Processing Letters,* vol. 24, pp. 279-283, March 2017.

[49] E. Koh, F. Saki, Y. Guo, C.-Y. Hung and E. Visser, “Incremental Learning Algorithm for Sound Event Detection,” in *2020 IEEE International Conference on Multimedia and Expo (ICME),* 2020.

[50] C. Mydlarz, S. Nacach, A. Roginska, T. H. Park, E. Rosenthal and M. Temple, “The Implementation of MEMS Microphones for Urban Sound Sensing,” *Journal of The Audio Engineering Society,* pp. 740-748, 2014.

[51] U. P. Fabra, “Freesound API documentation,” [Online]. Available: https://freesound.org/docs/api/resources_apiv2.html. [Accessed 20 March 2023].

Appendix 1 – Non-exclusive licence for reproduction and publication of a graduation thesis[1]

I Jelizaveta Kärner

1. Grant Tallinn University of Technology free licence (non-exclusive licence) for my thesis “Creation of Strongly-Labelled Dataset of Tallinn Urban Noise”, supervised by Jaanus Kaugerand
   1.1. to be reproduced for the purposes of preservation and electronic publication of the graduation thesis, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright;
   1.2. to be published via the web of Tallinn University of Technology, incl. to be entered in the digital collection of the library of Tallinn University of Technology until expiry of the term of copyright.
2. I am aware that the author also retains the rights specified in clause 1 of the non-exclusive licence.
3. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights, the rights arising from the Personal Data Protection Act or rights arising from other legislation.

02.01.2024

[1] The non-exclusive licence is not valid during the validity of access restriction indicated in the student's application for restriction on access to the graduation thesis that has been signed by the school's dean, except in case of the university's right to reproduce the thesis for preservation purposes only. If a graduation thesis is based on the joint creative activity of two or more persons and the co-author(s) has/have not granted, by the set deadline, the student defending his/her graduation thesis consent to reproduce and publish the graduation thesis in compliance with clauses 1.1 and 1.2 of the non-exclusive licence, the non-exclusive license shall not be valid for the period.