Henrique Seixas Moura

AUTOMATIC RECOGNITION OF BABY CRY

1 2 9 0 UNIVERSIDADE D COIMBRA

1 2 9 0

UNIVERSIDADE D
COIMBRA

Henrique Seixas Moura

AUTOMATIC RECOGNITION OF BABY CRY

Dissertation in the context of the Master in Informatics Engineering, Specialization in Intelligent Systems, advised by Professor Rui Pedro Paiva and Professor César Teixeira and presented to
Faculty of Sciences and Technology / Department of Informatics Engineering.

January 2022

Faculty of Sciences and Technology

Department of Informatics Engineering

Automatic recognition of baby cry

Henrique Seixas Moura

Dissertation in the context of the Master in Informatics Engineering, Specialization in Intelligent Systems advised by Prof. Rui Pedro Paiva and Prof. César Teixeira and presented to the
Faculty of Sciences and Technology / Department of Informatics Engineering.

January 2022

1 2 9 0

UNIVERSIDADE Ð
COIMBRA

This page is intentionally left blank.

Acknowledgements

The masters thesis marks the last challenge in a long arduous journey. I feel compelled to express my gratitude to the people who helped in any way throughout this year of learning and hard work and for making it much easier to accomplish. More specifically, I would like to give my thanks to all mentioned bellow:

To my advisors from the Department of Informatics Engeneering of the University of Coimbra, Professor Rui Pedro Paiva and Professor César Teixeira, for their availability, knowledge and guidance that directly elevated the quality of the work presented.

To my parents, who have always helped me focus on my work without ever putting pressure on me. One acknowledgment section is not enough to describe how lucky I feel for having their support.

To my friends, that were there whenever I needed to relieve stress or just to see faces that put me at ease, and to the ones that accepted to help me when I needed to test the mobile application developed in this work.

To Ana, who was a very important person that spent the most time fighting with me during this challenge and always had my best interest in mind.

Finally, to the one parent couple who took their time to believe in this project and help by sending baby cry recordings.

Abstract

Throughout time, decoding baby cry has been a challenge for parents and, more recently, for researchers in the field of pattern recognition. This thesis focuses on making progress on the field of pattern recognition, by exploring feature extraction from audio samples of baby cry, along with feature selection and reduction techniques to assess which features are deemed most valuable to have in a feature set, studying the performance of traditional machine learning approaches and building a database of baby cry.

To achieve these goals, an initial analysis was conducted to understand the state of the art regarding the machine learning approaches used and the current mobile application market, as well as studying the audio features that had a major impact in the feature extraction process. In addition, an Android mobile application named "BabyCry" was developed with the intent to create a sizeable annotated baby cry database. The idea was to distribute it to interested parents who would use it to record and annotate the cry of their babies and send it to a cloud database, namely Firebase. However, despite our hard recruitment efforts, we obtained only one acquisition. As such, we had to employ a publicly available dataset, the Baby Chillanto database, which contains 138 recordings of baby cry, on the following categories: hunger, pain, deafness and asphyxia; from which only the samples of hunger and pain were used. Another database named Donate-a-Cry was also used, containing cries of hunger, pain, discomfort, tiredness and eructation, however the results obtained had a low impact due to the evident class imbalance. From the samples of these databases, frequency, timbre and intensity features were extracted, which, after applying statistical analysis, resulted in a total of 882 features. As for the data collection, the distribution process did not go as planned due to a low adherence to the submission of audio samples, which resorted in a slight shift in plans for this work.

Afterwards, several classifiers were implemented, namely Support Vector Machines, K-Nearest Neighbours, Random Forest and Minimum Distance Classifier, and their performance was compared in a set of experiments, with the purpose of inferring the classifier that could deliver the best results more swiftly. In this experimental work, some feature selection techniques were applied, namely the removal of low variance features, the Pearson correlation and Minimum Redundancy Maximum Relevance algorithm, as well as the feature reduction technique called Principal Component Analysis, with the purpose of studying their impact. From the experiments performed, the best result was obtained by the SVM classifier with an RBF kernel, achieving a 78.08%±8.81% classification accuracy when fed 50 features extracted and selected from the Baby Chillanto database without the use of PCA. Similar good results were also obtained by the K-NN classifier when fed 21 features extracted, selected and reduced by the use of PCA from the Baby Chillanto database, achieving a classification accuracy of 78.03%±11.03% and the highest f1-score of pain cry of 73.34%±14.41%.

Previous studies have achieved better results in terms of accuracy using these classifiers. This can be justified by the fact that said studies had a larger private database, since they mostly conduct their own sample collection. When using the Donate-a-Cry database, it was also shown that studies may also unknowingly mislead by only showing the accuracy results, given that imbalanced datasets, as it was the case, tend to provide good accuracy results, yet if other metrics, such as the f1-score, are used, it can be seen that a model might be trained to only predict the majority class correctly.

Keywords

Baby Cry, Acoustic Analysis, Traditional Machine Learning, Feature Extraction, Feature Selection, Feature Reduction, Mobile App Development

This page is intentionally left blank.

Resumo

Ao longo do tempo, descodificar o choro de bebé tem sido um desafio para os pais e, mais recentemente, para investigadores da área de reconhecimento de padrões. Esta tese foca-se em fazer progressos no campo do reconhecimento de padrões, explorando a extração de atributos em amostras de áudio de choro de bebé, juntamente com técnicas de seleção e redução de atributos para avaliar que atributos são considerados mais úteis dentro de um conjunto, estudando o desempenho de abordagens tradicionais de machine learning e trabalhando na construção de uma base de dados de choro de bebé.

Para atingir estes objetivos, foi realizada uma análise inicial para entender o estado da arte das abordagens de aprendizagem computacional utilizadas e o mercado atual de aplicações móveis, assim como estudar os atributos de áudio que tiveram maior impacto no processo de extração de atributos. Além disso, foi desenvolvida uma aplicação móvel em Android chamada "BabyCry", com a intenção de criar uma base de dados de choro de bebé de tamanho considerável. A ideia era distribuí-la por pais interessados que a usariam para gravar e anotar o choro dos seus bebés e enviá-lo para uma base de dados na nuvem, chamada Firebase. No entanto, apesar de nossos árduos esforços de recrutamento, obtivemos apenas uma aquisição. Posta esta falta de dados, tivemos que utilizar bases de dados disponíveies publicamente, a base de dados Baby Chillanto, que contém 138 registos de choro de bebé, nas seguintes categorias: fome, dor, surdez e asfixia; das quais foram utilizadas apenas as amostras de fome e dor. Também foi utilizado outra base de dados denominada Donate-a-Cry, contendo choros de fome, dor, desconforto, cansaço e eructação, porém os resultados obtidos tiveram baixo impacto devido ao visível desequilíbrio de classes. Das amostras dessas bases de dados, foram extraídos atributos de frequência, timbre e intensidade, que, após a aplicação de análise estatística, resultaram num total de 882 atributos. Quanto à recolha de dados, o processo de distribuição não correu como planeado, devido à baixa adesão ao envio de amostras de áudio, o que levou a uma ligeira mudança de planos para este trabalho.

Posteriormente, foram implementados vários classificadores, nomeadamente Support Vector Machines, K-Nearest Neighbours, Random Forest e Minimum Distance Classifier, e o seu desempenho foi comparado num conjunto de experiências, com o objetivo de inferir o classificador que poderia produzir os melhores resultados mais rapidamente. Neste trabalho experimental foram aplicadas algumas técnicas de seleção de atributos, nomeadamente a remoção de atributos de baixa variância, a correlação de Pearson e o algoritmo Minimum Redundancy Maximum Relevance, assim como a técnica de redução de atributos denominada Principal Component Analysis, com o objetivo de estudar seu impacto. Das experiências realizadas, o melhor resultado foi obtido pelo classificador SVM com um kernel RBF, alcançando uma exatidão de 78,08%±8,81% quando lhe foi fornecido 50 atributos extraídos e selecionados da base de dados Baby Chillanto sem recorrer à PCA. Também se obtiveram bons resultados quando se usou o classificador K-NN quando lhe fornecido 21 atributos extraídos, selecionados e reduzidos pelo uso de PCA das amostras da base de dados Baby Chillanto, alcançando uma exatidão de 78,03%±11,03 % e o melhor f1-score de choro de dor de 73,34%±14,41%.

Estudos anteriores obtiveram melhores resultados em termos de exatidão usando estes classificadores. Isso pode ser justificado pelo facto de os referidos estudos possuírem uma base de dados privada maior, uma vez que a maioria coletou as próprias amostras. Ao utilizar a base de dados Donate-a-Cry, também foi demonstrado que os estudos também podem inadvertidamente induzir em erro ao mostrar apenas os resultados de exatidão, uma vez que bases de dados desequilibradas, como foi o caso, tendem a fornecer bons resultados de exatidão, mas se se usarem outras métricas, como o f1-score, pode-se observar que um modelo pode ser treinado para apenas prever corretamente a classe em maioria.

Palavras-Chave

Choro de Bebé, Análise Acústica, Aprendizagem Computacional Tradicional, Extração de Atributos, Seleção de Atributos, Redução de Atributos, Desenvolvimento de Aplicações Móveis

This page is intentionally left blank.

Contents

This page is intentionally left blank.

Acronyms

CNN Convolutional Neural Network. 15, 43

DBL Dunstan Baby Language. 8, 12, 13, 15, 28

DFT Discrete Fourier Transform. 26

FFT Fast Fourier Transform. 27, 33

K-NN K-Nearest Neighbours. iv, vii, 2–4, 13–16, 28, 30, 33, 35–39, 43

LFCC Linear Frequency Cepstral Coefficients. 28

LPC Linear Prediction Coding. 13, 15

MDC Minimum Distance Classifier. iv, vii, 2–4, 32, 33, 37, 38, 43

MFCC Mel Frequency Cepstral Coefficients. 3, 13–16, 26–28, 33, 35

MLP Multilayer Perceptron. 15

MRMR Minimum Redundancy Maximum Relevance. iv, vii, 29, 38

PCA Principal Component Analysis. iv, vii, 14, 29, 30, 36–38, 41

RBF Radial Basis Function. iv, vii, 14, 37

RF Random Forest. iv, vii, 2–4, 15, 16, 31, 33, 37, 39, 41–43

RNN Recurrent Neural Network. 15

STFT Short Time Fourier Transform. 25

SVM Support Vector Machines. iv, vii, 2–4, 14–16, 30–33, 35, 37, 39, 42, 43

ZCR Zero Crossing Rate. 15, 25

This page is intentionally left blank.

List of Figures

This page is intentionally left blank.

List of Tables

This page is intentionally left blank.

Chapter 1

Introduction

The act of crying is our first attempt at communicating how we feel to the world. Babies cry to draw the attention of their caregivers, which causes them to have an emotional and behavioural response with the intent of assessing the baby's discomfort and alleviate them. Being the most used form of vocal communication by babies, a cry can be interpreted in various ways. Therefore decoding the meaning of each cry is usually a problem for new parents that, although they eventually develop an aptitude to understand what is being communicated, may be left in frustration for wanting to aid their child but not knowing what might be distressing them. Furthermore, there have always been people with a certain skill to distinguish cries, however it can never be classified as a trustworthy or reliable method since the human perception in this matter is subjective.

In the present days, a baby is handled by not just the parents, but health professionals, other family members or day-care workers, emphasizing this need for understanding. Many people have tried to come up with ways of discerning baby cry by just using human hearing, and this matter has become a study subject in a plethora of fields such as paediatrics, psychology, psychiatry, neurology, etc. but, with the advances in computer science, it is increasingly becoming a scientific challenge in this field, as pattern recognition techniques and sound recognition technology are explored.

The decoding of baby cry goes beyond just helping parents. The benefits of understanding a baby's language range from just knowing if the baby is hungry or sleepy to a possible diagnosis of pathologies. Although it is still an area under research, there have been studies that correlated baby cry with certain illnesses including asphyxia and other respiratory problems, sudden infant death syndrome, Down syndrome, brain damage, etc. (LaGasse et al., 2005), which can be identified by analysing the cries of the baby suffering from these diseases. This grants a huge advantage in the medical field, allowing for an additional diagnosis support tool that may prevent an infant to suffer complications.

1.1 Objectives and Approaches

Although some studies have been performed with great advancements in the last 20 years, the subject of baby cry classification resorting to pattern recognition techniques is still fairly undeveloped. This can also be reflected on the available mobile applications that make use of machine learning algorithms to classify baby cry. Its scarcity and lack of complexity show that there have been attempts at building a reliable algorithm to recognize baby cry and integrate it in a smartphone application, yet the public still shows some dissatisfaction towards the products available in the market.

Given the importance and relevance of having an extra tool that possesses the ability of helping parents, health professionals and caretakers, this master's thesis aims to further the progress on the field of baby cry analysis, having two main goals. The first goal is to study and implement traditional classifiers and compare their performances, when discerning hunger cries from pain cries. The purpose of this primary goal is to assess the classifier with the best compatibility of being integrated into a mobile app with the functionality of swiftly classifying baby cry in real time.

In order to build any machine learning model, a dataset is required, which leads to the second goal of this work, which is the collection of baby cry samples. Currently, there is a lack of available baby cry datasets to the public, which makes these types of studies more complicated to accomplish. Taking that into account, the second goal of this thesis was to develop a mobile application capable of making recordings, labelling the cries and submitting them to a database where they could be processed.

For the first semester, the goal was to first perform a critical analysis of the state of the art regarding the problem of automatic baby cry detection to discern cries of pain from cries of hunger, comparing different classification techniques, and arrange an already available database of baby cries to perform classification experiments with Support Vector Machines. The second semester focused on building the baby cry recording app and its subsequent distribution to interested parents and further experimenting with other classifiers aside from Support Vector Machines.

1.2 Contributions

With the accomplishment of the goals set above in mind, this thesis contributed by firstly providing an up to date literature review in the field of baby cry, discussing the available and some of the private databases, the most recurrent classifiers used along with the most commonly extracted features and the results obtained for each study. A brief discussion of the current mobile application market is also provided, highlighting the most downloaded applications as well as their best features or lack of them.

For the data collection goal, even though the results obtained were not satisfactory in terms of quantity, a recording mobile app was built and it can still be used, making it easier to collect annotated samples and perform further work on this field.

Lastly, this thesis also showcases the performance of a selected group of traditional machine learning classifiers, namely Support Vector Machines, K-Nearest Neighbours, Minimum Distance Classifier and Random Forest, obtaining results that, in terms of accuracy, which is the most used metric in these studies, may have been slightly bellow other studies' results, however it is also hinted at the fact that accuracy may not be the only or best metric to use, and that, in order to have satisfactory results, there needs to be a large and balanced datset available.

1.3 Planning

The purpose of this section is to showcase the distribution of the work and the estimated time for each task for the first semester, represented in Figure 1.1, along with the actual time expended for each activity, represented in Figure 1.2. In addition, in Figure 1.3 it is also possible to see the estimated plan for the second semester and, in Figure 1.4, the tasks that were performed and their duration.

In the first semester, the first month was dedicated to reading the current literature on baby cry analysis and investigating what databases were used when creating machine learning classifiers. The second month's focus was on exploring traditional classifiers and getting familiarized with the code libraries used to extract features such as Mel Frequency Cepstral Coefficients. The elaboration of the State of the Art started in the third month along with the continuation of the exploration of the baby cry approaches. In the fourth month, the experiment with a traditional classifier and the study of the deep learning techniques used in baby cry classification were initiated. IT was also planned to implement deep learning techniques, however, due to the low amount of samples gathered from the databases available, this option was discarded, only to be pondered again if a sizeable database could be retrieved. Initially, the master's thesis plan encompassed the implementation of a mobile application capable of recognizing and classifying baby cry, additionally offering suggestions on why the baby was feeling that way and how to deal with specific cries. So, the fifth month was dedicated to analysing the requirements and architecture of the mobile application to be developed in the second semester, proceeding with the writing of the intermediate report and planning the work for the second semester. In the end of the first semester after 6 months of research, it was noted that in order to build such an application, a sizeable dataset would have to be used. With that in mind, the direction of this thesis had to be thought over, since there was no guarantee that such a dataset would be attainable, as it was later concluded.

In the second semester, the first two months were dedicated to the implementation and testing of a mobile application that would record baby cry. In the estimated plan this task was estimated to only last a month, yet as the application was developed, new features were taken into account, and some aspects suffered major changes with said additions, leading to an extended duration of the task. In the second month, there was already a first version of the application, therefore testing ensued in that period, until it a final deployable version was produced, as well as the writing of the thesis. In the third month, with the app finally ready, the distribution process could start and, while recordings were awaited, it was given continuation to the preliminary experiments from the first semester, which consisted of tweaking the experiment in terms of features that were being extracted, what recordings and evaluation metrics were being used. By the end of November, few recordings had been gathered, which resulted in the changing of the plan, mentioned in the end of the first semester. Since there was no data and no time to implement an automatic baby cry recognition application, the thesis proceeded in another direction, which aimed to test different classifiers, in order to assess which would be the fittest for a real time prediction application. So, after completing the adjustments to the experiment of the first semester using a Support Vector Machines classifier, the implementation of K-Nearest Neighbours, Minimum Distance Classifier, Random Forest classifiers ensued, as well as the comparative study of the results obtained from all the above.

Throughout planning in both semesters, some tasks ended up taking more time than estimated. In most cases, this was derived from poor time estimation, yet some occurrences, mainly on the second semester, derived from the fact that, since the initial classifier integrated mobile application was not developed, the other practical tasks were furthered, namely the experiments performed with the classifiers.

1.4 Outline

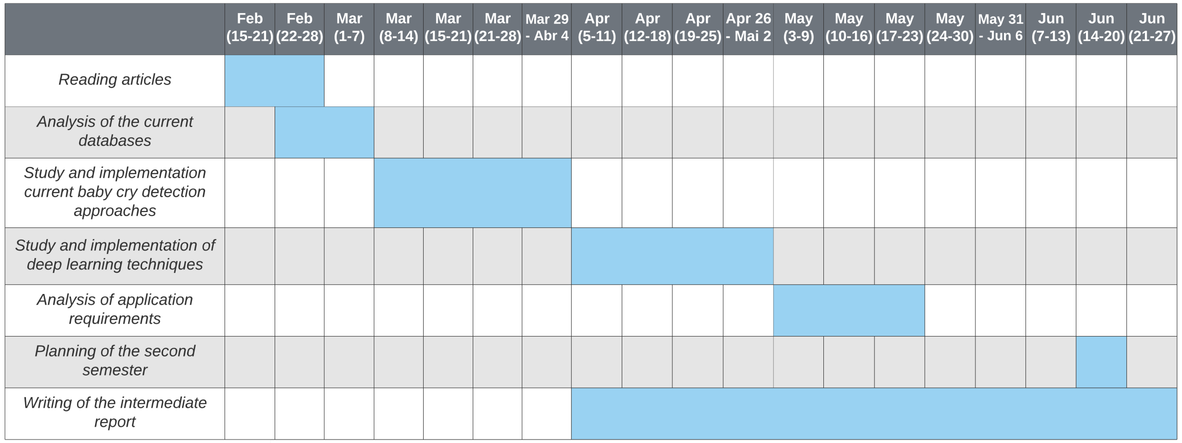
This document is divided into 5 chapters, each having multiple sections. The first chapter is the current one, where an introduction to this work is delivered. The purpose of this chapter is to give the reader a notion of what the motivation was to perform this work, the goals that were set and how it was planned to achieve them and what was actually possible to achieve and the final contributions generated.

The following section of the document (Chapter 2) firstly covers the context of this work along with some basic concepts, with the purpose of filling the reader on the psychological and acoustic aspects of baby cry e.g. what the fundamental frequency and formants are and how they impact the reading of a cry. Next is delivered the state of the art, where the most used databases in the field of baby cry are described along with the current approaches used in relevant studies with said databases. In this chapter, the current available mobile applications capable of automatic baby cry recognition are also discussed, presenting an overview on their features and available information as well as possible drawback.

Chapter 3 which describes the efforts made in the annotated baby cry collection field, by reporting the recording app development process, portraying its architecture and progress achieved throughout the development and testing, ending with a discussion of the achieved outcome.

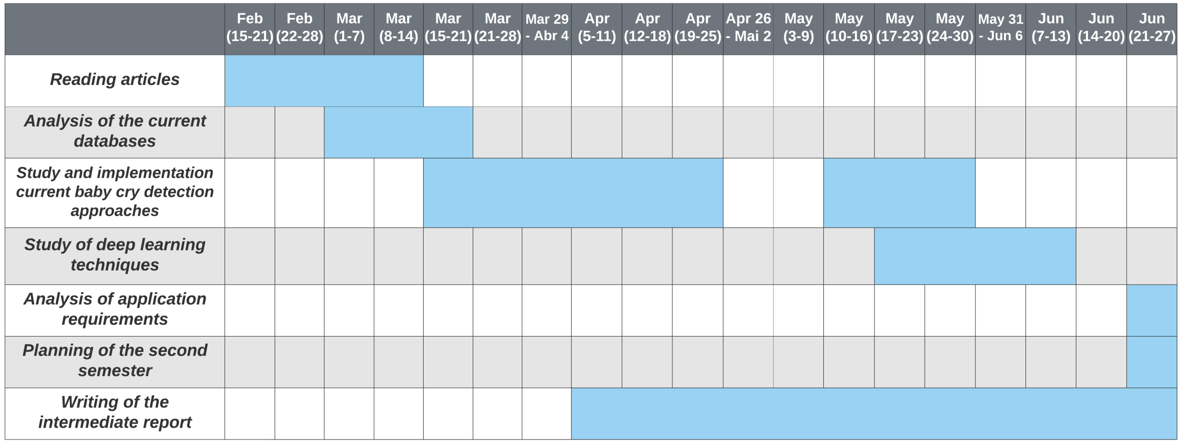
Chapter 4 covers the bulk of this work, starting with an explanation of feature extraction in audio files is presented along with a description of the feature selection and reduction techniques that were used in this work. The next section of this chapter presents an introduction to traditional machine learning techniques that were considered for the experiments that ensued. Which leads to the next and final section of this chapter, which mainly reports the experiments performed, where several comparative studies were made, assessing the performance of Support Vector Machines, K-Nearest Neighbours, Minimum Distance Classifier and Random Forest classifiers along with the testing of two different available databases named Baby Chillanto and Donate-a-Cry.

To close this document, Chapter 5 offers a recap on the experiments performed and the overall content of the thesis, additionally presenting a possible direction for the continuation of this work in the future.



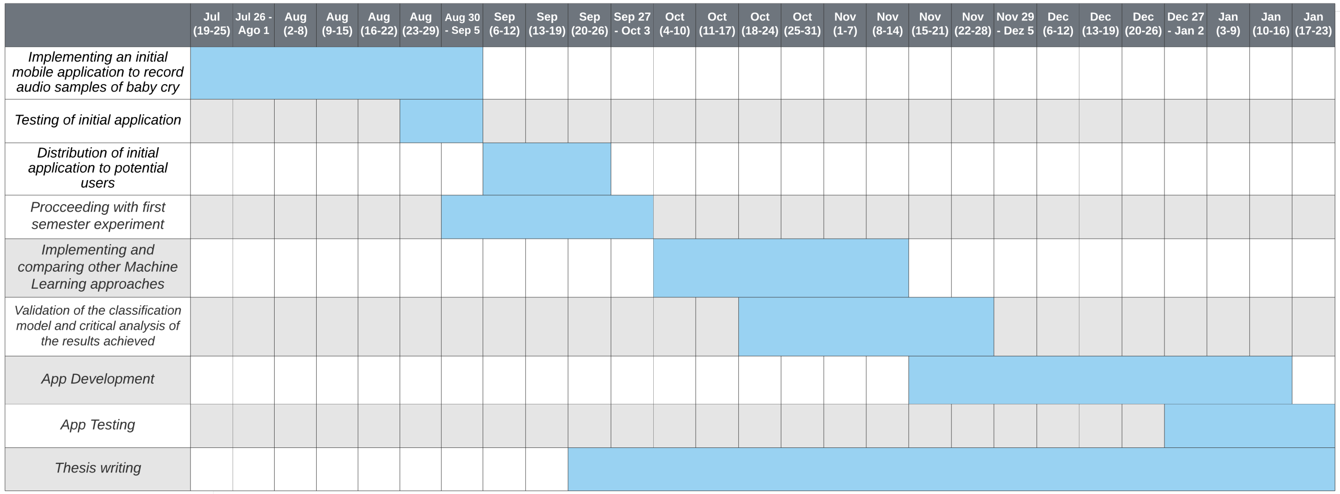
	Feb (15-21)	Feb (22-28)	Mar (1-7)	Mar (8-14)	Mar (15-21)	Mar (21-28)	Mar 29 - Abr 4	Apr (5-11)	Apr (12-18)	Apr (19-25)	Apr 26 - Mai 2	May (3-9)	May (10-16)	May (17-23)	May (24-30)	May 31 - Jun 6	Jun (7-13)	Jun (14-20)	Jun (21-27)
Reading articles	■	■																	
Analysis of the current databases		■	■																
Study and implementation current baby cry detection approaches				■	■	■	■												
Study and implementation of deep learning techniques								■	■	■	■								
Analysis of application requirements												■	■	■					
Planning of the second semester																		■	
Writing of the intermediate report								■	■	■	■	■	■	■	■	■	■	■	■

Figure 1.1: Gantt Diagram - Expected Plan for the First Semester



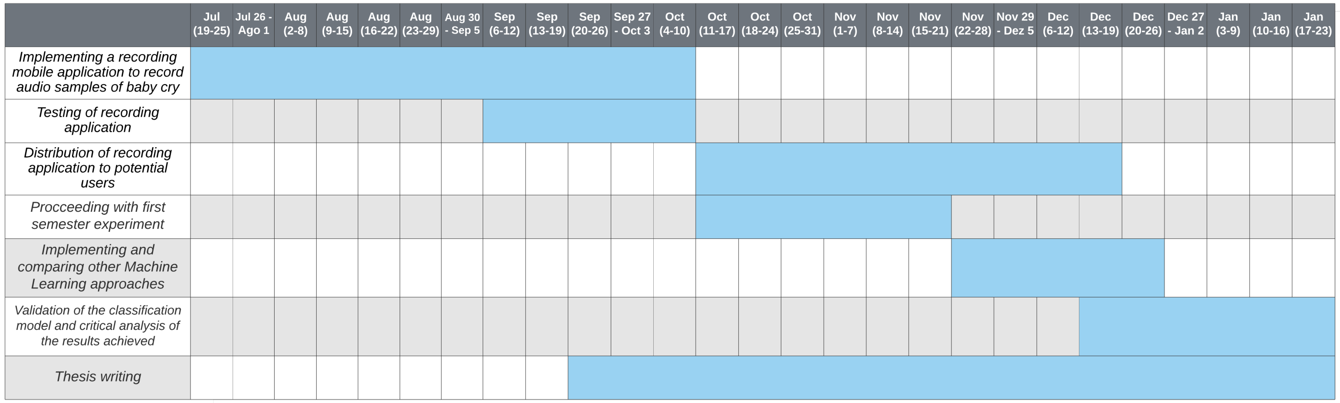
	Feb (15-21)	Feb (22-28)	Mar (1-7)	Mar (8-14)	Mar (15-21)	Mar (21-28)	Mar 29 - Abr 4	Apr (5-11)	Apr (12-18)	Apr (19-25)	Apr 26 - Mai 2	May (3-9)	May (10-16)	May (17-23)	May (24-30)	May 31 - Jun 6	Jun (7-13)	Jun (14-20)	Jun (21-27)
Reading articles	■	■	■	■															
Analysis of the current databases			■	■	■														
Study and implementation current baby cry detection approaches					■	■	■	■	■	■			■	■	■				
Study of deep learning techniques														■	■	■	■		
Analysis of application requirements																			■
Planning of the second semester																			■
Writing of the intermediate report								■	■	■	■	■	■	■	■	■	■	■	■

Figure 1.2: Gantt Diagram - Real Plan for the First Semester



	Jul (19-25)	Jul 26 - Ago 1	Aug (2-8)	Aug (9-15)	Aug (16-22)	Aug (23-29)	Aug 30 - Sep 5	Sep (6-12)	Sep (13-19)	Sep (20-26)	Sep 27 - Oct 3	Oct (4-10)	Oct (11-17)	Oct (18-24)	Oct (25-31)	Nov (1-7)	Nov (8-14)	Nov (15-21)	Nov (22-28)	Nov 29 - Dez 5	Dec (6-12)	Dec (13-19)	Dec (20-26)	Dec 27 - Jan 2	Jan (3-9)	Jan (10-16)	Jan (17-23)
Implementing an initial mobile application to record audio samples of baby cry	■	■	■	■	■	■	■																				
Testing of initial application						■	■																				
Distribution of initial application to potential users								■	■	■																	
Procceeding with first semester experiment							■	■	■	■	■																
Implementing and comparing other Machine Learning approaches												■	■	■	■	■	■										
Validation of the classification model and critical analysis of the results achieved														■	■	■	■	■	■								
App Development																		■	■	■	■	■	■	■	■	■	
App Testing																								■	■	■	■
Thesis writing										■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

Figure 1.3: Gantt Diagram - Expected Plan for the Second Semester



	Jul (19-25)	Jul 26 - Ago 1	Aug (2-8)	Aug (9-15)	Aug (16-22)	Aug (23-29)	Aug 30 - Sep 5	Sep (6-12)	Sep (13-19)	Sep (20-26)	Sep 27 - Oct 3	Oct (4-10)	Oct (11-17)	Oct (18-24)	Oct (25-31)	Nov (1-7)	Nov (8-14)	Nov (15-21)	Nov (22-28)	Nov 29 - Dez 5	Dec (6-12)	Dec (13-19)	Dec (20-26)	Dec 27 - Jan 2	Jan (3-9)	Jan (10-16)	Jan (17-23)
Implementing a recording mobile application to record audio samples of baby cry	■	■	■	■	■	■	■	■	■	■	■	■															
Testing of recording application								■	■	■	■	■															
Distribution of recording application to potential users													■	■	■	■	■	■	■	■	■	■					
Procceeding with first semester experiment													■	■	■	■	■	■									
Implementing and comparing other Machine Learning approaches																			■	■	■	■	■				
Validation of the classification model and critical analysis of the results achieved																						■	■	■	■	■	■
Thesis writing										■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

Figure 1.4: Gantt Diagram - Real Plan for the Second Semester

This page is intentionally left blank.

Chapter 2

State of the Art

Even though there have always been experts trying to decode an infant's cry using solely the human ear's capabilities, in the last fifty years there has been an increasing development of pattern recognition tools to analyse speech which led to an increase of infant cry studies making use of such tools and consequently developing specific cry detection techniques. In this chapter, the core concepts are discussed along with the databases and techniques that have been mostly used in this field.

2.1 Context and Background Concepts

Infant cry is described as an acoustic manifestation composed of vocalization, constrictive silence, coughing, choking, interruptions, or various combinations of such sounds. Through these acoustic manifestations alone it is possible to collect a high amount of physical and psychological information about the baby, regarding the health, weight, gender, diseases, and emotions (Mittal et. al, 2014).

Crying can be classified as not just an infant behaviour but as an integral part of the assurance of the human species survival, since it lets others know how helpless a human baby is, eliciting them to attend their distress, consequently putting them both in a state of strong sympathetic nervous system activation as described as a "Fight or Flight" response. Caretakers or bystanders can take a "flight" response, by avoiding the baby and trying to distance themselves, as it can be witnessed when an infant is crying in public places such as a supermarket or restaurant, where people commonly avoid contact with the crying baby. People can also take a "fight" response, which should be the more adequate response if the person in question is the baby's caretaker, since it should be of their best interest to calm the infant down by assessing the possible problem and cease their distress (LaGasse et at., 2005).

However, attending to a crying infant can be a challenging task, especially for new parents and even other more experienced caretakers in the sense that it is not easy to discern what each cry means. Since it is important to truly understand how a new-born is feeling through their only way of vocal communication, several studies have been conducted to try and understand them. One example is the Dunstan Baby Language (DBL), suggested by Priscilla Dunstan, an Australian musician with the talent to memorize all kinds of sounds, who classified infant cries into five different baby languages: hunger, represented by the sound "neh"; tiredness, represented by "owh", which indicates when the baby is getting sleepy; eructation, meaning the need to burp which is represented by "eh"; pain in the stomach, represented by "eairh" and general discomfort represented by "heh" (Dunstan P., 2012). These cries are considered to be normal cries, meaning, they are the usual types of cry a baby will express. There is also a different category labeled pathological cries, for babies with more serious complications such as respiratory, neurological, heart, digestive or infectious issues.

Baby cry is an acoustic signal with strong harmonic content generated by using a rapid flow of air through the larynx, resulting in a burst of sound that consequently makes the vocal folds open and close which generates periodic excitation. This excitation is transferred through the vocal tract to produce a high-pitched sound which generally has a fundamental frequency (F_0) or pitch of 250-600Hz (Bhagatpil et al., 2014). Figure 2.1 illustrates the anatomy of cry, displaying which parts are involved in its production and propagation. In a vocal emission, there are two important characteristics worth mentioning: the fundamental frequency and the formants.

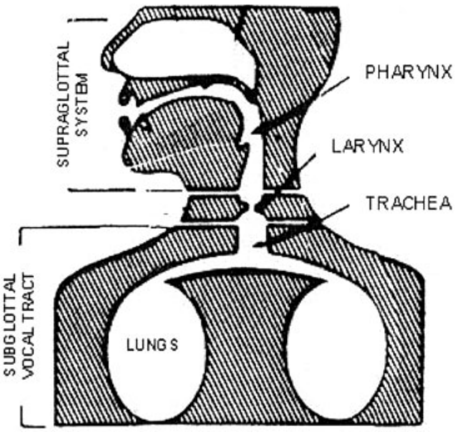


Figure 2.1: Anatomy of Cry (LaGasse et al., 2005)

The fundamental frequency can be defined as the result of the median frequency in Hz of the vocal fold vibration, present at the top of the trachea, heard as voice pitch. When these vibrations have a source on the glottis they are known for their quasi-periodicity. Voiced sounds like vowels, semi-vowels and nasal sounds are examples of quasi-periodic vibrations. Unvoiced sounds on the other hand come from the turbulence on different parts of the vocal apparatus (Fort, A. & Manfredi C., 1998). From the F_0, it is possible to extract relevant information about a new-born's wellbeing. Brain malfunctions or instabilities can affect the laryngeal coordination and consequently the glottis vibration, making it noticeable when F_0 is analysed. In addition, infant cry is sometimes described as a rising-falling melody, characterized by an initial increase of F_0 followed by a final decrease. Abnormal duration of these two phases along with the frequency of F_0 can be indicative of brain pathologies (Fort, A. et al., 1996).

A cry can also be classified into three different categories depending on the F_0. The first two compose voiced sounds, being phonation, which is characterized for cries that have an F_0 of 400Hz-600Hz, and hyperphonation, characterized by an F_0 of 1000Hz-2000Hz, which is the result of a qualitatitive vocal shift in vocal production and is normally an indication that the infant has suffered from a prenatal condition that may have compromised their neurobehavioral organization (Zeskind, P., 2011). The third category is labeled disphonation which is another definition of unvoiced sounds, characterized for possessing a non-harmonic nature, being noisy and turbulent. These types of cry can also be an indicator of various pathologies such as asphyxia, hyperbilirubinemia, gastroschisis, and respiratory distress syndrome (Abbs, K. J., 2015).

The formant frequencies, also termed resonance frequencies, are a result of the acoustic resonance of the vocal tract. These frequencies, just as F_0, are shaped by the supraglottal system which is controlled by the brainstem. Normally only the first (F_1) and second (F_2) formant are calculated, the first representing frequencies centred at the first resonance of F_0 and the latter representing frequencies centred at the second resonance of F_0 (LaGasse et al., 2005).

2.2 Databases

Over the years, several baby cry databases have been proposed in the literature, with different dimensions, taxonomies, target populations, signal recording quality, and, consequently, qualities. In fact, the quality of a baby cry dataset can be assessed based on the following factors:

- **Recording:** the type of recorder used and the proximity of said recorder to the baby can influence the quality of the audio in the sense that if the recorder is too far away it might not pick the sound of the baby well enough to discern it afterwards.
- **Background noise:** a realistic database should have environmental sounds, e.g., people talking in the background, environment music, etc.
- **Taxonomy:** the dataset should comprise a broad range of different cries and not only a single cry. For example, Raina P. et al. focus on analysing a single type of cry, making it not as useful since the goal is to use a dataset that is comprised of the most matching parameters needed. In particular, hunger and pain are two key cries we aim to address in our study, hence these two must be present. Ideally, the five categories identified by Priscilla Dunstan should be present.
- **Population:** some databases may have multiple types of cry but then lack in quantity. In this case, an ideal dataset should have enough information, either in number of cries or in total recording time, to extract discerning characteristics from each type of cry. In addition, there are other factors that can influence the dataset such as the age range of the babies being recorded.
- **Availability(Public/Private):** when choosing a dataset, there is also the big possibility that the data is restricted and unavailable for other future studies or worse, the dataset does not exist anymore. The database must be public, so that different researchers can employ it and benchmark each other works.

Furthermore, it needs to be noted that these datasets tend to be curated by a person who is in charge of annotating the cry, i.e., filling out the information regarding it. This makes room for some flaws in the dataset, given that the impact of subjectivity, errors and missing information in the annotation process are usually more prevalent when annotation agreement between different annotators is not followed, which deteriorates the quality and veracity of the data being analysed.

In Table 2.1 are represented some of the datasets that were already used in baby cry studies along with their key properties, which will be discussed further regarding why (or why not) they are fitting for this work.

Table 2.1: Databases Overview

Name	Source	Population	Recording Setup	Background Noise	Cry Causes Analysed	Availability
IIIT-S ICSD Cry	Pranaam Hospital, India	Age: 3 months to 2 years Number of cries: 693 Total duration: 670.1s Health status: Normal	Roland R-09 Wave/MP3 Stereo Sampling Rate: 48KHz Coding Rate: 24 bits	People present were requested to remain silent during recording	Pain, Discomfort, Emotional Need, Ailment, Environmental Factors, Hunger/Thirst	Requestable
Dunstan Baby Language	Dunstan Baby, Australia	Age: Newborn to 3 months Number of cries: 82 Total duration: 362 Health status: Normal	Sampling Rate: 44.1KHz	Unspecified	Flatulence, Eructation, Discomfort, Hunger, Tiredness	Private
SPLANN	Sf. Pantelimon Emergency Clinical Hospital, Romenia	Age: Newborn to 3 months Number of cries: 18473 Duration: 30-50s clips Number of Babies: 136 babies Health status: Normal and Pathological (respiratory, neurological, heart, digestive, infectious, genetic syndromes, etc.)	microphone iRig Mic connected to a Samsung Galaxy S4; CryingPicker	Hospital and Home environment	Colic, Eructation, Discomfort, Hunger, Tiredness, Pain, Pathology	Unspecified
Raina et al.	Jehangir Hospital, India	Age: Newborn to 1 month Duration: average 6s clips Number of babies: 100 babies Health status: Normal and Pathological (Unspecified)	Sony ICD-UX200F stereo IC recorder Stereo Sampling Rate: 44.1KHz Coding Rate: 16 bits	Peaceful environment with minimal external disturbances	Normal, Pathological	Unspecified
Baby Chillanto	INAOE-CONACyT, Mexico	Age: Newborn to 9 months Number of cries: 138 Total duration: 2274s Health status: Normal, Deaf and Asphyxiating	Mono Sampling rate: 8KHz, 11.025KHz, 22.050KHz	Unspecified	Asphyxia, Hunger, Deaf, Pain, Normal	Requestable
Donate-a-Cry	Github gveres/donateacry-corpus User uploaded	Age: Newborn to 2 years Number of cries: 457 Total Duration: 2742s Health status: Normal	Mobile application for Android and iOS Sampling Rate: 8KHz Uniform bit rate: 128kbps	White noise, baby chatter and adulti mimicking noises have been removed	Hunger, eructation, colic, discomfort, tiredness,	Public

The Infant Cry Sounds Database from the Indian Institute of Information Technology (IIIT-S ICSD) is comprised of data collected by the Pranaam hospital, Madinaguda, Hyderbad, where a total of 693 infant cries were collected from babies aged 3 months to 2 years for a total of 670.1 seconds of audio. The cry signals were extracted in routine check-ups, vaccination trips or when the baby was in any emotional need of attention, and background noise was minimal, so the people present in the room were asked to maintain silence during the recordings. Said recordings were performed in stereo mode with a Roland R-09 Wave/MP3 recorder, placed at 10-20 cm from the baby, with a sampling rate of 48GHz and 24 bit coding rate. The main causes of infant cry encountered where pain, either caused by vaccination, physical hurt or internal pain, discomfort, emotional need, ailment, such as feeling too cold, coughing or a feverish and hunger/thirst. This dataset can mostly be seen in research papers by Vinay Kumar Mittal (2014, 2015a, 2015b, 2016a, 2017) where it is stated that the features extracted consisted of the magnitude of the short-time Fourier transform spectrogram and the fundamental frequency and harmonics. We requested the authors the dataset and after being granted access it was noticeable that the crying samples lacked labelling, making it impossible to know which baby cry meant. After contacting the institution regarding this issue they replied that the person responsible for the labelling was no longer a part of the project and that there was no way to retrieve it therefore rendering this database useless.

The Dunstan Baby Language (DBL) database has been used differently by several studies due to its diverse information and its trustworthiness. It is a collection of cries from 1000 infants of up to 3 months old in seven countries and a total of 30 nationalities, proving once again how the five baby languages proposed by Priscilla Dunstan are universal regardless of the place of birth. Despite being a renowned database, little information can be found about it, besides having sampling frequencies of 8KHz, 16KHz and 44.1KHz and analysing the expected five types of cry: Flatulence, Eructation, Discomfort, Hunger and Tiredness. This database would be ideal to have access to, due to having labelled types of cry that are relevant to this work and permission to use it was requested, however access was denied.

The SPLANN database is part of a project developed by a software development company named SOFTWIN with the help of the Faculty of Electronics, Telecommunications and Information Technology "Politehnica" University of Bucharest and the Emergency Clinical Hospital "Sf. Pantelimon" that aimed to design and develop an automatic infant crying recognition system, using signal processing and pattern recognition techniques. In said hospital recordings of a total of 136 newborns up to 3 months old were performed acquiring 13373 cries. The recordings took place inside seven different workstations, each equipped with a unidirectional microphone iRig Mic mounted on a tripod placed at 20cm from the infant's mouth and it was connected to a Samsung Galaxy S4 smartphone by jack. There was also given the possibility for the parents to record at home, using a dedicated android application called CryingPicker, which recorded about 5100 cries (Rusu et al., 2015). As stated previously, although this possibility allows for more data extraction, it also allows for some data misclassification since it is up to the parents to mention which type of cry their baby is expressing. In total, seven different types of cry were analysed: hunger, pain, eructation, tiredness, discomfort, colic, which were considered normal cries, and pathological cries. Home recordings had an average duration of 30 seconds and hospital recording had an average duration of 50 seconds. Unfortunately, no information on the availability of this dataset was obtained.

The Baby Chillanto is a continuously growing database from the National Institute of Astrophysics and Optical Eletronics, CONACyT, Mexico, currently comprised of 138 different cries for a total of 2274 seconds. After contacting the Universidad Autonoma de Tlaxcala, it was possible to have access to this dataset and explore the data in a more in-depth manner. First thing to be noticed was that the database was fully labelled, containing five types of cry: hunger, pain, normal, deafness and asphyxia. Although deafness and asphyxia are pathological cries and do not have much importance for this work specifically, the hunger and pain cries can be useful to replicate past studies made with this database and used for further developing of the work. Each cry type had a folder with the full clips of audio and another with one second segments of those clips. It was also noticed that the sampling rate of the clips varied from 8KHz, 11.025KHz and 22.050KHz.

The Donate-a-cry corpus was the second database that was accessible for analysis, since it is currently a GitHub repository dedicated to storing user-uploaded audio samples of infant cry under the Open Database License. Its original purpose was to be used as part of a final project in Speech Technology Course in the Royal Institute of Technology in Sweden. These audio samples are uploaded through the use of a dedicated mobile app and, due to their raw nature, they have to be checked and modified to become part of the final dataset which is the one that will be used. This database contains 457 different cries, where a large part of them (387) are hunger cries, each having the duration of 6 seconds, for a total of 2742 seconds of audio in wav format with uniform bit and sampling rate of 128kbps and 8KHz respectfully. It is important to reference that given that the clips are uploaded by people in the community, there is some issues with this dataset such as the recording quality or mislabelled of cries.

2.3 Current Approaches

Using the DBL database, for feature extraction, a large part of researchers make use of the Munich open Speech and Music Interpretation by Large Extraction, also known as openSMILE (Tuduce et al., 2018). This tool allows extraction of chroma features, MFCC features, Fundamental Frequency, voicing probability, loudness features and Emotion feature set. The Emotion feature set used is named emobase and contains a carefully selected set of 1582 features out of the 6552 features present in another Emotion set called emo_large. The reason for this preference of a smaller set of features is justified by researchers who found it more accurate (Parlak et al, 2104) . Another feature set worth mentioning is the COMputational PARalinguistics ChallengE (ComParE), that is comprised of more features than the aforementioned (over 6125 features), enabling for a more complex feature selection, although some of the features are not needed in baby cry analysis, consequently raising computation cost as the size of the database in question increases.

Studies that have used the SPLANN dataset have once again made use of the openSMILE tool, more specifically, using the extraction tool ComParE (6373 features) along with MFCC and F_0 (Tuduce et al., 2019).

Several studies have made use the Baby Chillanto database, recurring mostly to the feature extraction of MFCC. Rosales-Pérez et al. (2015) extracted features of MFCC and LPC and used a fuzzy model for a classifier to discern the types of cry available in the dataset. One of these experiments was specifically to discern cries of pain from cries of hunger where it was possible to obtain a 97.96% classification accuracy. The first goal with this dataset is to experiment with the extraction of MFCC combined with the use of SVM to create a classifier that is able to successfully distinguish cries of pain from cries of hunger.

Bano et al. (2015) made use of the K-NN classifier by feeding it features such as pitch frequency, short-time energy, harmonicity factor, harmonic-to-average power ratio and MFCC extracted from recordings performed in a hospital to classify the five baby languages suggested in DBL. It is also stated that each type of cry had 50 samples, 40 for training

Table 2.2: Baby Cry Classification Approaches

Authors	Cries Analysed	Databases	Features/Input	Classifiers	Best Performance
Sahak R. et al.	Normal and Asphyxia	Baby Chillanto	MFCC	SVM	Accuracy: 95.86%
Maghfira et al.	Hunger, pain, eructation, discomfort and colic	DBS	Spectrogram	CNN and RNN	Accuracy: 94.97%
Franti et al.	Hunger, pain, eructation, discomfort and colic	DBS	Spectrogram	CNN	Accuracy: 89%
Bano et al.	Hunger, pain, eructation, discomfort and tiredness	Self-Recorded	MFCC	KNN	Accuracy: 80%-90%
Orlandi et al.	Term and Pre-Term	Self-Recorded	F0, F1, F2 and F3	Logistic Regression, MLP, SVM, Random Forest	Accuracy: 87%
Tejaswini et al.	Hunger, pain and discomfort	Self-Recorded	MFCC	SVM	Accuracy: 93.1%
Silva et al.	Pain, Screaming, Yell moan, Frustration, Upset	Day-cares, neighbours and online databases	MFCC	KNN	Accuracy: 81.67%
Onu et al.	Normal and Asphyxia	Baby Chillanto	MFCC	SVM	Sensitivity: 85% Specificity: 89%
Manikata et al.	AC & fan, cry, speech, music in the background	Home recordings	MFCC	1D-CNN, FFNN, SVM	F1-score: 98.86%
Kulkarni et al.	Hunger, tired, eructation, discomfort and belly pain	Donate-a-Cry	MFCC, LPC and spectral features	KNN, RF, SVM	Accuracy: 84%

and 10 for testing and that for the distance metric algorithm the Euclidean Distance was chosen, reaching a correct detection of 80% of the cries of hunger and sleepiness, and 90% of the cries of pain, colic and discomfort.

Silva et al. (2017) also made use of the K-NN classifier and MFCC and pitch frequency as features extracted from several databases such as day-care centres, neighbours and online databases and compared the use of the classifier primarily in MATLAB and later the code was converted to C and tested in a Raspberry Pi board. With 150 samples for training and 120 for testing the maximum accuracy obtained was 81.67% which once again is not a very satisfactory which leads to the assumption that K-NN by itself may not be enough to obtain satisfactory results. It is also stated that the combination of MFCC with pitch frequency showed more promising results than using just the MFCC. Finally, unrelated to the classifier, this study made an interesting observation about cry detection on a device, stating that the accuracy results obtained were slightly lower than the results obtained in MATLAB. Since the end goal of this work is to have a mobile app we might also experience a decrease of accuracy when testing it on the phone.

Onu et al. (2017) developed and mobile application with the functionality of discerning normal cry sounds from asphyxia cry sounds using MFCC extracted from the Baby Chillanto dataset, and a SVM classifier which used 80% of data for training and validation and 20% for testing. Although no accuracy values are disclosed, it is mentioned that they were able to achieve sensitivity results, meaning the ratio of correctly classified asphyxia cries, and the specificity results, meaning the ratio of correctly classified normal cries, of 85% and 89% respectively. Sahak et al. (2016a) experimented discerning once again the normal cries from the asphyxia cries from the database Baby Chillanto, this time testing the use of SVM, and SVM combined with Principal Component Analysis (PCA), using a Linear and Radial Basis Function (RBF) Kernel and feeding it the MFCC extracted. The training and datasets were divided randomly by using 5-fold cross validation and, in order to find the optimal hyperplane, a Quadratic Programing algorithm was used. The most satisfactory result obtained was with the combination of a SVM with a RBF Kernel with PCA, achieving an accuracy of 95.86%.

Tejaswini S. & Natarajan S. (2016) created a baby cry classifier, recurring to Support Vector Machines, to discern baby cries of pain, hunger and discomfort. The dataset used consisted of baby cry recorded at M S Ramaiha teaching and memorial hospitals, Banglore, India. Wavelet transform was applied and Mel Frequency Cepstral Coefficients were extracted. The multi-class problem was divided into three binary one-vs-one classification problems, obtaining a classification accuracy of 90.27% when discerning pain cries from hunger cries, 71.29% when discerning disconfort cries from pain cries and the best result of 93.095% when discerning hunger cries and discomfort cries.

Orlandi et al. (2016) performed a comparative study of four different classification methods: Logistic Curve, Multilayer Perceptron, Support Vector Machines and Random Forest. The problem consisted in classifying baby cry into two classes, those being full-term and preterm baby cries, in order to analyse the differences between them and enhance the chances of survival of preterm and very low weight neonates. With a self-recorded dataset consisting of more than 3000 cries from 28 full-term and 10 preterm newborns, features were extracted through the use of a software dedicated tool developed at the Biomedical Engineering Lab, University of Firenze, Italy, called BioVoice, and twenty two acoustical parameters were estimated, consisting of the statistical modelling of F_0 and the formants F_1, F_2 and F_3. There were three different testing options considered: full training set, 10-fold cross validation and 66% split. The results obtained pointed that the classifier that achieved the best performance was the Random Forest, with about 87% accuracy when using 10-fold cross validation.

Franti et al. (2018) applied CNN to discern the five baby languages from DBL by feeding the network the spectrograms obtained from the infant cry audio to process it as an image. The training and testing sets, consisting of 250 and 65 cries respectively, were selected randomly but equally distributed within the five classes, being able to obtain an accuracy of 89%. Later, Maghfira et al. (2020) analysed the results obtained by Franti et al. (2018), and reproduced the experiment but this time using 5-folds cross validation and a combination of Convolutional Neural Networks and Recurrent Neural Networks and were able to achieve an accuracy of 94.97% when using the CNN-RNN combination with cross validation.

Manikanta et al. (2019) analysed deep learning approaches as well as a machine learning approach, under indoor background sound environments, those being: one-dimensional Convolutional Neural Network, Feed Forward Neural Networks and multi-class Support Vector Machines. The study consisted in discerning Ac and fan noises, music, speech and baby cry recorded, with a sampling rate of 44.1KHz, under various home conditions, from which were extracted Mel Frequency Cepstral Coefficients. The experiment was divided in three different frame lenghts of audio, 100ms, 250ms and 500ms, which had overall optimal results, the best result being an f1-score of 98.86% when using the one-dimensional Convolutional Neural Network with a frame length of 500ms.

Audio is a form of sequenced data, therefore, the use of RNN on infant cry detection has been scarce but nevertheless has shown good prediction results. Maghfira et al. (2020) justifies the addition of RNN to CNN based on the logic that, since sequences of audio properties through time can change the predictor's decision, infant cry audio needs to be analysed from the beginning to the end, otherwise we take a risk of misclassification due to the missed information in the middle of the audio.

Kulkarni et al. (2021) performed a study comparing the performance of traditional classifiers such as K-Nearest Neighbours, Support Vector Machines, Random Forest and logistic regression and how individual features and the combination of all features influenced the classification statistics. The study used the public dataset Donate-a-Cry from which were extracted features such as Mel Frequency Cepstral Coefficients, Linear Prediction Coding, spectrall flatness, roll-off, centroid, flux, bandwidth, Zero Crossing Rate, gamma tone frequency cepstral coefficients, . When testing individual features both the K-NN and Random Forest provided the highest classifications, yet when testing with all features combined the best result was obtained with the Random Forest classifier. In addition, the study also showed that the gamma tone frequency cepstral coefficients lead to a better performance than the MFCC.

After analysing the different approaches, it was possible to note the most relevant features being used, i.e. MFCC and fundamental frequency, etc., and the classifiers that were chosen most frequently and delivered the best results. With that in mind, it was of interest to perform experiments were said classifiers, such as K-NN and SVM, could be analysed and possibly attempt to emulate the results obtained in studies that used the same databases available to this work, namely Kulkarni et al. (2021).

2.4 Current Mobile Applications

Currently it can be said that the market for a reliable mobile application that is able to recognize and classify baby cry is still fairly unexplored. Although there are already some apps on the market, they show signs of being underdeveloped and flawed, leading to people not yet considering them reliable enough for daily use. This derives mainly from the difficulty of gathering eligible samples to form a strong database, since baby cry can have a variety of acoustic differences depending on the sex and age of the baby. From the app store, four apps were installed in order to understand how they operated, their strong points and their weaknesses, named Babba, Baby Language, Cry Analyzer and ChatterBaby. In this section each app will be analysed individually.

ChatterBaby was developed by a team at UCLA with the initial main goal of helping deaf parents to understand their infants' needs and then generalized to creating an app that could help any parent. The project built its own database that is updated every time a person submits a new recording to be classified. This recording is sent to the servers where unwanted background noise is removed and an algorithm that is not specified returns the baby cry type predicted to the smartphone. Aside from a recording menu, the app has several other menus that only contain information available on the ChatterBaby website. After recording an audio the app will first predict if in fact the audio corresponds to a baby cry. If so it will offer up to three of the best predictions each with a percentage of certainty from 14 different preset cry types. In case the app fails to predict a cry successfully it will also suggest that the user inputs their opinion on what the correct cry type was, where they can select from the 14 types of cry or input a new type of cry.

Babba is a korean app that allows for recordings of up to 10 seconds, which are sent to a server where the algorithm used returns one of 5 cry types, those being hunger, need of diaper change, tiredness, eructation or other. The app firstly tries to detect if the audio is from a crying infant and then suggests the top two cry types that may be occurring. In addition, the app also shows some acoustic information and suggests how the parents should proceed to alleviate their crying infant by showing specific tips of each cry type. Unlike ChatterBaby, this app has the limited free use of three predictions per day, meaning that, if a user wants to use the app more regularly, they will have to pay a subscription to this service. This paid subscription also allows a user to have access to an improved version of the app that eliminates white noise for a better prediction. In terms of resourcefulness, Babba may be the safest option for a parent.

Baby Language is an app created by a software developer who, after being a parent for the first time, started to develop it by himself in 2013 and later on with a team of contributors on the field of translation and design. The app allows for the recognition of 5 different types of cry, those being hunger, eructation, belly pain, tiredness and irritation, through the use of an embedded speech recognition toolkit from CMU-Sphinx, meaning this app will not require internet connection to work as intended. However, the app comes only with a limited trial and to use the full version, a single payment is required. The full version also allows the user to have access to tips on how to recognize the specific cry type, how to act and even how it can be prevented.

Lastly, Cry Analyzer was developed by a Japanese company called First-Ascent, whose main business model is to create mobile applications. The app is fairly simple, and, after filling out some information about the baby on the first use, such as the age, gender and location, is composed of three menus. The first is the recording menu where a user can record an audio and send it to the company's servers where an algorithm will process it and return the two most likely cry types, which can be any of the following 5: hunger, tiredness, boredom, irritation and discomfort. The next menu is a history of previous predictions where the user can input a cry type, in case they think the app did not perform correctly. This app also comes with a limited trial, so, if a user wants full access, they need to pay a one time fee. The third menu is the settings menu, where a user can check information about the app, the data inputted on the first opening of the app and billing information.

Chapter 3

App Development

With the data gathered on the first semester it is possible to perform experiments to test which features are more useful and what classifier provides the best results, however, in order to make a reliable classifier embedded in a mobile application, it is necessary to gather more recordings. To do so, a recording app with the name "BabyCry" was created to be used by parents of babies of up to two years of age, preferably younger than a year old. These age limitations arise from the culmination of two factors, those being that a baby's vocal tract changes significantly in two years, resulting in different vocalizations and because after the 1-year mark, babies start to express themselves with words, meaning that crying is no longer the main way to communicate what they are feeling. In this chapter, the development of this mobile application, built to satisfy the second goal of this thesis of collecting annotated samples of baby cry will be reported.

3.1 Methodology

For the development of this mobile application, a Scrum methodology was adopted, given that it is a methodology fit to work well with small teams that focuses on the project and on constant improvement via feedback from the advisors, when attending regular weekly meetings, testers and people who helped throughout every stage of this part of the project. Due to the constant changes that had to be done, it was considered that a methodology like Waterfall was not suited for this type of work, since said methodology required the definition of rigorous processes where each task can only be started once the previous one is completed. The Lean methodology could also have been applied, however it was discarded due to the need for beforehand careful planning combined with the fact that initially this app was built to be temporary, while recordings were being gathered, and then it would have to be reworked into also being an app capable of performing the classification of recordings. With all the above considered, Scrum was chosen to be the best methodology applicable. Each week focused on various aspects of the application and, at every meeting with the advisors, the work would be explained, possible changes would be discussed and the next step in the developing process would be planned for the next week.

3.2 Architecture

The goal of this application was to deliver it to as many people as possible, so, the operative system chosen was Android, having the back-end code written in Java and the frontend code written in XML. In terms of functionalities, the application was built to allow recordings of up to twenty seconds where, after that time passes or a user presses the recording again, the user can select what type of cry was expressed. Following the Dunstan Baby Language, it was decided that the user can choose from five predefined cry types: hunger, pain, eructation, tiredness and discomfort. In addition, in case the user wants to annotate a different cry type they can also select the option of inputting a cry type by selecting the option “other”. After labelling, the recording is sent to a folder in the external storage of the mobile phone and the referring information of the recording is stored in an SQLite database. This database has four relevant fields, those being a string containing the type of cry and its number, a string with the device ID of the smartphone and later it was added a string with the time and date of the recording and a cloud status column, added after the cloud storage feature was successfully implemented, to check if the recording had already been submitted.

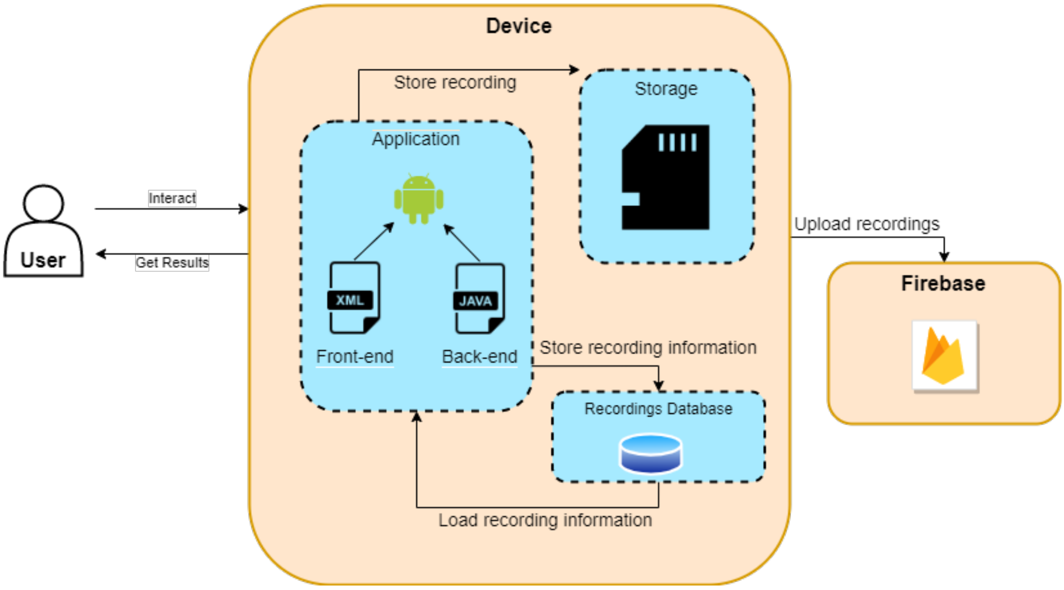


Figure 3.1: Application Architecture

Given that this mobile recording application is meant to receive recordings from multiple devices, the files had to be named in a manner where each file has a unique name, in order to facilitate the discerning of the recordings from each device. To achieve this, initially, each file was named as it follows “####CryType_DeviceID”, where the first 4 characters are the number of the recording, ranging from 0001 to 9999, followed by the cry type selected in the alert dialog that appears after finishing a recording, and finally the DeviceID which is a unique ID comprised of numbers and letters. In version two, with the addition of the date and time to the information of each recording, the filename was also changed to contain this information so it can be read as such: “####CryType_Date-Time_DeviceID”. In version three, the final field added to the filenames were the sex and age of the baby, achieving the final file name structure: “####CryType_Date-Time_SexAge_DeviceID”. In the edit filename option, the user can only change the “CryType” field, whereas the remaining fields are filed automatically. The underscores that separate each field after the CryType help process the different parts that comprise the entire filename, so when a user tries to input a new cry type, the underscore is part of a list of reserved characters that cannot be used along with prohibited characters on general filenames. For example, "0001Hunger_13112021-213032_F10_d44eaab93a64aa2ac1f4f0764ca412bdc38a961d331070d86" means that the file contains the first cry of hunger in the mobile device, which was recorded on the 13th of November of 2021 and expressed by a 10 month old female baby, from the device with the id expressed after the last underscore.

In order to gather the recordings from each device, a connection with the google cloud storage platform Firebase was implemented. By selecting the option to submit to the database in the application, the user can send the recordings to the Firebase storage. In addition, a user can also edit the cry type of the submitted recordings, which will change the name of the file in the external storage, the SQLite database and the Firebase storage, the latter requiring internet connection.

3.3 App Progress

Throughout the versions of the app, besides the recurrent help of the advisors, feedback was always requested to a small group of people outside the project in order to not only have a functional app but also to deliver an application that has an appealing design and is practical to use.

This small group of people consisted of friends and family members, within the age groups of 20-30 and 50-60 years old respectively, who were willing to take a part of their time to review the application. Regarding the backgrounds of the people selected, two are worth mentioning given that their background was taken into consideration when selecting them and those were a graphical designer and a informatics engineer.

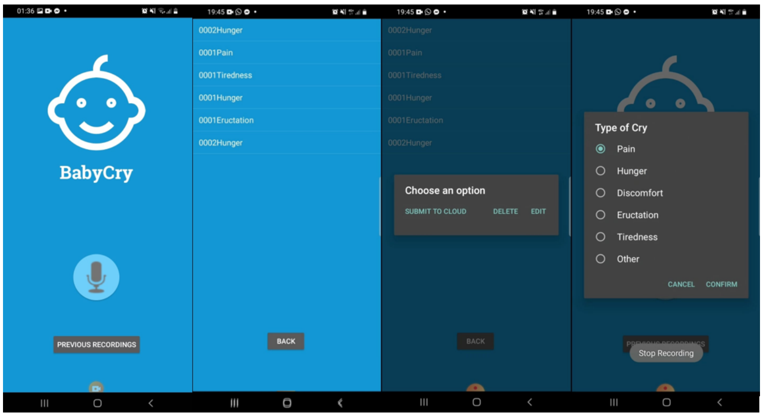
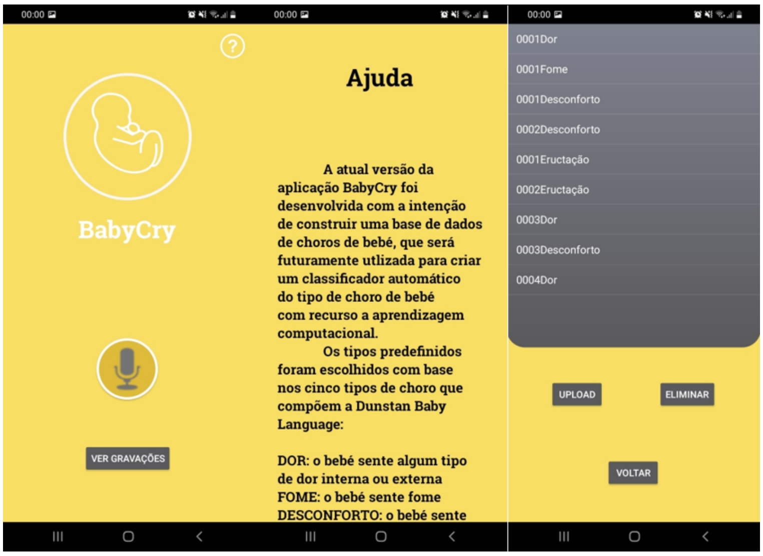


Figure 3.2: First version of the app

Figure 3.2 showcases the first version of the app. In this version the base functionalities were implemented, although they were still in a very primordial state, with the exception of the option of saving the recordings in the database. This version allowed the recording and device storage of annotated baby cry, as well as the ability to visualize and manage recordings. In the recordings menu it is possible to edit the name of a recording or to delete it. In the following versions it was taken into account that this recording app would be used by parents living in Portugal, therefore the language of the app was changed to Portuguese to facilitate its use.

After discussing possible improvements and new features with the advisors, a second version of the app was created, as it can be seen in Figure 3.3. In this second version, the functionality of storing recordings in the database was implemented, which was achieved by resorting to Google's cloud service Firebase. Although the key functionalities of the app were implemented there was still some features that needed to be present in order to provide a pleasant experience to the user. As it was said, this version allowed cloud storage, however it was not prepared to edit the label of submitted recordings. This feature



Ajuda

A atual versão da aplicação BabyCry foi desenvolvida com a intenção de construir uma base de dados de choros de bebé, que será futuramente utlizada para criar um classificador automático do tipo de choro de bebé com recurso a aprendizagem computacional.

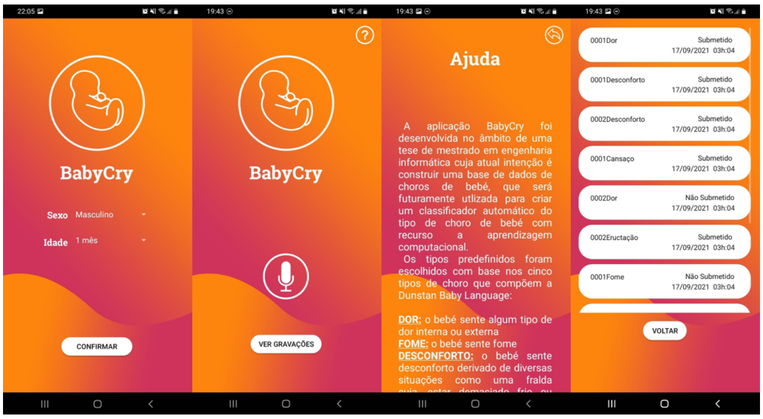
Os tipos predefinidos foram escolhidos com base nos cinco tipos de choro que compõem a Dunstan Baby Language:

DOR: o bebé sente algum tipo de dor interna ou externa
FOME: o bebé sente fome
DESCONFORTO: o bebé sente

0001Dor
0001Fome
0001Desconforto
0002Desconforto
0001Eructação
0002Eructação
0003Dor
0003Desconforto
0004Dor

Figure 3.3: Second version of the app

is important, since a user can submit a recording they are unsure of and later change the label, therefore allowing a more accurate classification.



Sexo Masculino
Idade 1 mês

Ajuda

A aplicação BabyCry foi desenvolvida no âmbito de uma tese de mestrado em engenharia informática cuja atual intenção é construir uma base de dados de choros de bebé, que será futuramente utlizada para criar um classificador automático do tipo de choro de bebé com recurso a aprendizagem computacional.

Os tipos predefinidos foram escolhidos com base nos cinco tipos de choro que compõem a Dunstan Baby Language:

DOR: o bebé sente algum tipo de dor interna ou externa
FOME: o bebé sente fome
DESCONFORTO: o bebé sente desconforto derivado de diversas situações como uma fralda

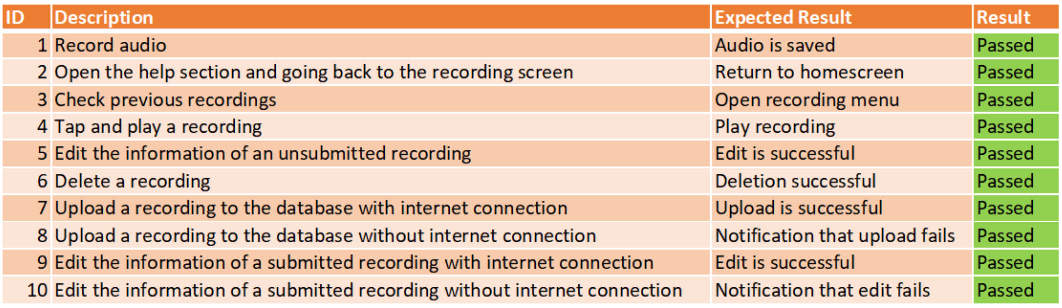
0001Dor Submetido 17/09/2021 03h:04
0001Desconforto Submetido 17/09/2021 03h:04
0002Desconforto Submetido 17/09/2021 03h:04
0001Cansaço Submetido 17/09/2021 03h:04
0002Dor Não Submetido 17/09/2021 03h:04
0002Eructação Submetido 17/09/2021 03h:04
0001Fome Não Submetido 17/09/2021 03h:04

Figure 3.4: Third version of the app

Finally, in Figure 3.4 it is showcased the final version of the app before it was distributed amongst parents. This version was created to correct the problems of the last version and to add other features, like the ability to edit the label of a submitted recording, more information of said recording when in the managing recordings menu, such as the time and date of the recording and an indicator of whether the recording has been submitted to the cloud storage or not. In this stage of development, another important aspect was taken into consideration. As seen in the first semester, the sex and age of the baby significantly affect the vocalization of their cries. With that in mind, an initial screen of the app was created, with the purpose of being displayed the first time a user opens it, where the sex and age of the baby are inquired, to allow for a more specified classification. This information was also added to the filename of each audio.

3.4 Usability Tests

After implementing the final version of the app, before distributing it to parents, several usability tests were issued to be performed by the selected testers, to assure all aspects of the application were working as intended and to get user feedback and possibly find areas of improvement. The first step of the usability tests was to list every task that needed to be performed. In total each tester was asked to perform a total of 10 tasks, as it can be seen in Figure 3.5.



ID	Description	Expected Result	Result
1	Record audio	Audio is saved	Passed
2	Open the help section and going back to the recording screen	Return to homescreen	Passed
3	Check previous recordings	Open recording menu	Passed
4	Tap and play a recording	Play recording	Passed
5	Edit the information of an unsubmitted recording	Edit is successful	Passed
6	Delete a recording	Deletion successful	Passed
7	Upload a recording to the database with internet connection	Upload is successful	Passed
8	Upload a recording to the database without internet connection	Notification that upload fails	Passed
9	Edit the information of a submitted recording with internet connection	Edit is successful	Passed
10	Edit the information of a submitted recording without internet connection	Notification that edit fails	Passed

Figure 3.5: Usability tests

The app was delivered to the testers along with an informative document with the tasks listed in Figure 3.5 and the usability tests ensued. Ideally, the usability tests would be performed with the tester in the same room, however due to the current pandemic, some of the testers performed the tasks by themselves, reporting afterwards what errors may have occurred or their opinion on what could be improved.

In terms of functionalities, the testers reported they encountered no problems aside from the way the application handled actions that required internet when there was no internet connection. After requesting to perform a submission of a recording without internet connection, it was noticed that the submission would be sent to the database as soon as there was internet connection, however the application would inform that the recording was already submitted even if there was no connection. Although this issue did not cause any conflicts, it was altered to stop the submission if there was no internet connection and show a message informing that the submission did not go through and that internet connection was necessary for that step.

In terms of usability, the testers reported that they found the application fairly straightforward aside from the submission process. To make a submission to the database, it is necessary to go to the list of previous recordings and long press the recording a user wants to submit and select that option in an alert dialog that appears. The main complain on this topic was that the action of long pressing on the recording to submit a recording was not very intuitive. Some testers suggested even that the submission option should be removed and, instead, every time a recording was issued, it would be automatically submitted. This option was pondered however it was later discarded since a user can record an audio that unknowingly contains sensitive information that they do not want to upload, so having the option of first hearing the recording and then submitting it was considered an important functionality to keep. Instead, an explanation of the submission process was added to the help section along with the creation of a document that explains the purpose of this project and how to use the application correctly, which will be discussed ahead.

3.5 Application Distribution

After testing the application, it was time to distribute it to parents. The initial idea was to distribute the application to parents that had babies with less than 2 years of age, preferably babies that were younger than 1 year old, and, throughout the next 2 months of this work, while the recordings were being gathered, the study and implementation of classifiers would be resumed. Finally, after those 2 months, the gathered recordings would be used to primarily perform experiments with the classifiers already implemented, and then to integrate the classifier that delivered the best results with the mobile application that allowed for baby cry recognition.

The first phase was to contact day-cares, kindergartens or parents individually, to ask if there would be any interest in participating in this work. The contacts were established in the cities of Leiria and Coimbra, in various ways, such as going to said establishments in person to talk to a representative that could help, phone calling and emailing the schools. After making the first contact, the school establishments that replied asked for more instructions, and, after seeing an appeal, a document was written with the intent of explaining what this work consisted of, the institution that was behind it and how the application worked, along with contact information in case of further questions or doubts. This document was sent to a google drive folder along with the android installation file for the recording app, and a link to that folder was sent to the establishments and people that were interested in participating.

Unfortunately, although interest was expressed by different parties, only one person was able to submit any recordings of baby cry, which is not enough to form a trustworthy dataset. This can have a multitude of justifications, namely, due to the current events, it was not easy to get a hold of parents personally, since some schools remained closed for most of the duration of this work and, even when open, there is a certain apprehension when establishing contact with people. The fact that most contacts had to be done via email or telephone combined with parents having little availability to help, given that parenthood is a difficult and arduous task especially in the first years of age, also created an understandable barrier.

Chapter 4

Baby Cry Analysis

In this chapter, the experimental part of this work will be discussed. Given the lack of samples gathered with the mobile recording application "BabyCry", no usable database could be concocted and, therefore, it was not possible to perform experiments with said database, having only the BabyChillanto and Donate-a-Cry databases available, which also have a reduced number of samples. Due to these reasons, it was opted to only use traditional machine learning techniques and to build them offline, instead of having an algorithm running in a server.

4.1 Feature Extraction

After collecting and preparing the data gathered in Section 2.2 it now needs to be transformed into features for further analysis. Features are attributes and specific traits of an object that help discern it from the rest of the objects. This initial process of feature creation is significantly important since the rigorousness of the acquisition of features can deeply affect the quality of the information retrieved.

In this work, feature extraction will be performed on the cry sounds collected and the features will be represented in a feature vector due to their resourcefulness since they facilitate the comparison of documents and structure a dataset which can be useful for data visualization, however it still has to be mentioned that having such a compact representation can cause loss of information. After retrieving the features, it is necessary to transform them into a reduced number of values in order to gain some insight from the data. This is called feature integration and usually consists on applying statistical modelling, making use of seven common statistics: the mean, standard deviation, skewness to measure the symmetry of the features, kurtosis to measure peakedness, median, maximum and minimum.

With this new vector of only 7 features, the range of values still varies widely which might make affect the performance of distance metrics. To solve this issue it is possible to apply feature scaling which consists in normalizing said range of values by either normalizing a variable to a specific interval, usually being [0, 1] or [-1, 1], called Min-Max Normalization, or transforming every variable to zero mean and unit standard deviation, called Standardization or z-score normalization.

Having a high amount of features is not always beneficial. Some features have less importance than others and some may be redundant, which may affect the classification performance. In addition, as the number of features increases, the data becomes sparser, creating a dimensionality problem. There is therefore a need to discard irrelevant features and to reduce the dimensionality of the data. With this goal in mind, the next step is to perform feature selection, which consists in applying techniques that solve these issues. It can be done manually, where a user decides the relevancy of the features by assigning weights to the features and the features with zero weight are discarded. It can also be done automatically by using dedicated algorithms such as Pearson Correlation to verify which features have a high correlation and then the user can delete the redundant ones, Kruskal-Wallis analysis to compute the rank of the features by their discriminant power and discard the ones ranking the lowest. To tackle the dimensionality problem specifically, other algorithms such as Principal Component Analysis for unsupervised feature reduction and Linear Discriminant Analysis for supervised feature transformation.

4.1.1 Core Concepts

The feature extraction process, as it can be expected, varies with the type of data being extracted. In this section the most used methods of audio feature extraction in the speech domain will be discussed. To begin this section, we will have to define some concepts starting with Short Time Fourier Transform (STFT). STFT is a time-frequency analysis method that works by applying Fourier Transforms on time segments of the signal, called epochs, instead of applying it to the whole signal. The reason for this segmentation comes from the fact that audio is time-varying, meaning that an audio signal has several different frequencies throughout its duration, but if with the use of time window, the signal is more likely to contain similar frequencies and becomes easier to analyse. The result obtained from this transformation are several magnitude spectrums where it is possible to see the presence and how powerful a frequency is in a signal. STFT works by having a user defined window that will slide through the signal with an overlap the user also defines. This may raise issues since if the size of the window is too small, few frequencies will be observable in each window, resulting in lower spectral resolution, yet choosing a big window size can result in the loss of temporal precision due to the integration of a longer period of time. As for the overlap, the bigger the overlap the higher the computational costs will be.

4.1.2 Types of Features

When analysing a file in audio form, there are a few known transversal concepts to capture everything that makes a sound discernible from the others, such as the frequency, the intensity and the timbre.

Frequency Features

For frequency features we can extract the Fundamental Frequency and the Zero Crossing Rate (ZCR). F_0, as stated before, is defined as the lowest harmonic present in a (pseudo-)periodic waveform, being more easily calculated for problems with monophonic sounds than for polyphonic sounds. The ZCR is the rate at which a signal changes from positive to negative or from negative to positive, that is know to be easy to compute. In a way, it serves as a noise measure, where the higher the noise present in an audio signal, the higher the ZCR (Giannakopoulos et al. 2014).

Intensity Features

For intensity we can extract features by measuring the energy of the signal over a window, called Root-Mean-Square Energy. Furthermore, we can calculate the mean RMS and calculate the percentage of frames bellow the mean to obtain the Less-Than-Average

Energy.

Timbre Features

For timbre features we can extract an abundance of features that will be shortly described with the exception of Mel Frequency Cepstral Coefficients (MFCC) due to its relevancy in this work. First we have the spectral features such as the spectral flux, which is a measure the amount of spectral change in a signal; the spectral flatness, which is a measure of the flatness of the magnitude spectrum, where the higher the flatness the higher the amount of white noise; the spectral rolloff, referring to the extraction the frequency, R, bellow which 85% of the magnitude distribution is concentrated giving information about the skewness of the frequencies being analysed; the spectral centroid, referring to the gravity of the magnitude spectrum, and the spectral bandwidth, which gives the broadness of the magnitude spectrum. The most used features however are the Mel Frequency Cepstral Coefficients. Its preference comes from the quality of the results that can be obtained given their ability to accurately represent the speech amplitude spectrum.

4.1.3 Visualizing a Cepstrum

Before mentioning MFCC, the Cepstrum must be defined. A Cepstrum is an anagram of the word spectrum which can be thought of as an interpretation to the mathematical transformations that are applied to a spectrum. It can be seen in Figure 4.1 that, to visualize a cepstrum, a signal must undergo several stages of processing. Starting with the waveform in the time domain, the first step is to apply a Discrete Fourier Transform (DFT) to obtain a power spectrum where the x-axis represents the Frequency and the y-axis represents power which shows how much each frequency is present in the original signal. The next step is to apply a logarithm to the power spectrum, obtaining a spectrogram of a continuous signal where the x-axis still represents the Frequency but the y-axis now represents Magnitude in dB. In this signal it is possible to observe a form of periodicity which is due to the original signal itself having harmonic components. With that in mind, for the final step, an inverse Fourier transform is applied with the purpose of finding which frequencies compose the signal. The result of this final transformation is called a cepstrum, where the y-axis represents Absolute Magnitude, and the x-axis represents a pseudo-frequency termed Quefrency with unit of reference being time e.g., milliseconds. In this cepstrum it is possible to see certain peaks in some frequencies, which represent harmonic sounds present in the signal.

4.1.4 Mel Frequency Cepstral Coefficients

First introduced in 1980 by Davis and Mermelstein, MFCC can describe the spectral shape of a sound using a perceptual frequency scale called the Mel scale. The human perception of pitch is non-linear, in the sense that, for example, if there were two sound samples, one with two low frequencies separated by X Hertz and another with two high frequencies separated by X Hertz as well, a human being would have more trouble distinguishing the sample with the higher frequencies than the sample with the lower frequencies. Therefore, human perception is faulty in this matter and a way of accurately perceiving sound needs to be implemented. To solve this issue, the unit of pitch, Mel Scale, was proposed by Stevens and Volkmann (1937). The Mel scale is a logarithmic scale where equal distances on the scale have the same perceptual distance. From 0Hz to 1000Hz the scale can almost be said to be linear, once again due to our more accurate perception of lower frequencies.

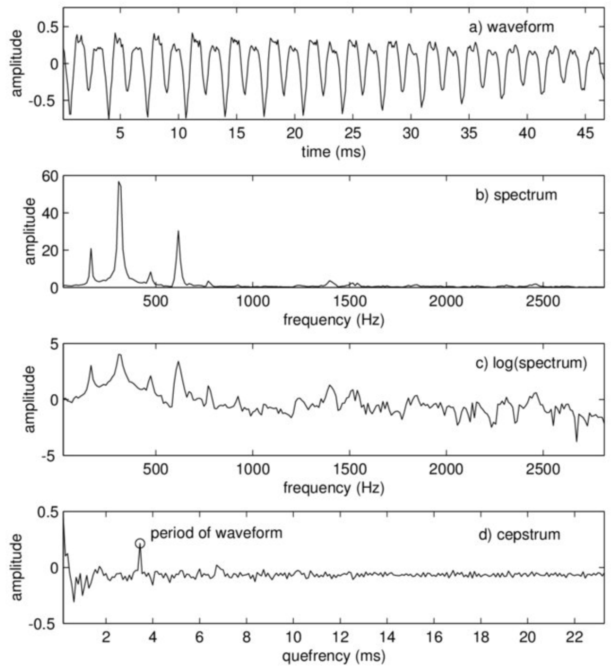


Figure 4.1: Visualizing a Cepstrum (Gehard, 2003)

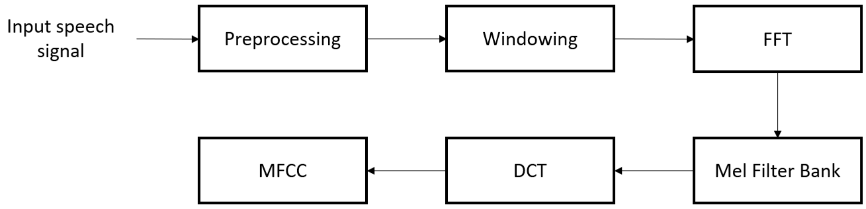


Figure 4.2: Process of obtaining Mel frequency cepstral coefficients

The algorithm behind MFCC can be divided into several steps, displayed in Figure 4.2. It starts by first segmenting the signal into frames, again due to the time-variance present in an audio signal, and applying Fast Fourier Transform (FFT) to each frame to obtain magnitude spectrums. These magnitude spectrums still contain a lot of unnecessary information so, in order to remove it, they are filtered with the Mel frequency filter bank to have a notion of how much energy is present in the multiple frequency regions. Typically 40 triangular filters are used, and in this step we define the centre frequency and lower and upper limits (in Mel) of each frequency for each filter and finally convert the filter frequencies back to Hertz. After applying the filters to the magnitude spectrums, we sum the filtered magnitude values within the limits of each filter, obtaining, in this case, 40 filter bank coefficients. As it was stated previously, humans do not hear loudness on a linear scale and for that reason we calculate the base 10 logarithm of these filter bank coefficients to match human hearing. The result obtained are overlapping filter banks which makes the filter banks energies be highly correlated with each other. The purpose of the final step is to decorrelate these values by calculating the DCT of the log filter banks energies, obtaining decorrelated coefficients called MFCC.

Additionally, from MFCC we can extract delta and delta-delta features, or in other words, differential and acceleration features. The need for these features comes from the fact that, although MFCC give important information such as formants and the spectral envelope of each frame, there is no continuous information on how it changes overtime which may be relevant in the case of speech to better understand the dynamic of the cry. The delta is calculated by taking the MFCC information of each frame and subtracting the values of the previous one and similarly, the delta-delta is calculated by using the delta values calculated of each frame and subtracting the previous one. Usually, these coefficients are added to the MFCC to successfully increase the performance of machine learning algorithm they are fed into.

4.1.5 Linear Frequency Cepstral Coefficients

Linear Frequency Cepstral Coefficients (LFCC) is another widely used method for speech analysis that has a similar feature extraction process to MFCC, with the exception that it uses a linear filter-bank instead of a Mel filter-bank. Sita et al. (2019) performed a study of baby cry analysis comparing the use of MFCC and LFCC combined with K-NN with the Euclidean Distance algorithm as the distance metric and Vector Quantization. It was concluded that a combination of LFCC with K-NN provided better results than any combination of the two classifiers with MFCC, with an average accuracy of 90% when discerning between crying and non-crying audio and an average accuracy of 90.83% when discerning between the five types of DBL. Although, when using MFCC, an accuracy superior to 80% was achievable, LFCC were indisputably a best choice in this scenario and this can be justified not because MFCC are a worse method in general but because the frequencies being analysed were high and LFCC tend to perform better with high frequencies and the Mel filter-bank is not the best choice for such task. The study also adds that when analysing female cry was relatively harder to classify correctly, due to their vocal tract being relatively shorter, producing formants of higher frequency, when compared to male infant cry.

4.1.6 Feature Selection and Reduction

As mentioned previously, a higher number features does not necessarily mean that the classification accuracy, in this case of an audio sample, is going to be improved. Therefore, it is important to assess the lowest amount of features that can deliver the best accuracy. In order to do that assessment, it is necessary to analyse how the features vary throughout the samples, how they correlate with each other and with the target classes as to understand which features influence negatively or do not influence at all the classification accuracy. In this section, the concepts from the field of feature selection and feature reduction used in the experimental work will be introduced.

Feature Variance

Feature variance refers to how much a feature varies throughout all samples. If a feature displays a high variance of values throughout the samples, it can be assumed that said feature may have a high influence on what the sample gets classified as. If a feature does not vary much or at all, it means the feature does not weigh significantly on the classification outcome, meaning it can only hinder the classification of a sample. When analysing extracted features, verifying their variance is a common first step to understanding which features can be removed without degrading the classification accuracy.

Pearson Correlation

In addition to assessing the feature variance, the Pearson correlation coefficient can be calculated in order to measure the correlation between features. Each feature is measure to have a correlation value between -1 to 1. If two features have a high correlation, whether it is a negative or positive correlation, it is implied that these features have a strong relationship, meaning their values regarding the class don't vary independently, therefore having both features in a feature set or just one of them will not affect the classification. Choosing which feature is removed can be done by checking the correlation values between both features and the target class, where the feature with the highest correlation with the target class is removed.

Minimum Redundancy Maximum Relevance

When ranking features based on their correlation with the class, picking the top-ranking features does not mean the classification accuracy that is going to be obtained is maximized. This is due to the fact that features can be highly correlated among themselves. In other words, if a feature is ranked highly, features possessing a high correlation with said feature will also be ranked highly, raising what is called redundancy of the feature set (Dunstan , 2012).

Redundancy can cause a decrease in efficiency, in the sense that, if a feature set is comprised by the x best features, yet half of those features are highly correlated with each other, this means that only the other half is truly representative of said feature set and eliminating the highly correlated features would not impact its classification. Therefore, when selecting features, it may be beneficial to not pick the x best features but the best x features.

The Minimum Redundancy Maximum Relevance (MRMR) approach ranks features using minimum redundancy criteria such as maximized mutual Euclidean distance or minimized pairwise correlations, along with the criteria that said features also have maximum relevancy to represent the entire dataset, by selecting features with maximal mutual information or dependence with the target.

For discrete variables, the minimum redundancy is calculated through maximizing the conditions of two equations, those being the mutual information to infer the dependency between features, and mutual information to infer the dependency between features and the target class. Two conditions can be generated from the two equations, named Mutual Information Difference and Mutual Information Quotient.

For continuous variables, the criterion functions are a combination of the F-statistic between features and the score of their maximum relevance. From this combination two criterion functions can be obtained, those being F-test Correlation Difference and F-test Correlation Quotient.

Principal Component Analysis

After applying feature selection, a feature set may still contain a large amount of complex information that is hard to visualize. Principal Component Analysis (PCA) is a multivariate statistical analysis approach used to combat this problem by reducing the dimensionality of the data. This dimension reduction comes with the cost of losing information and consequently accuracy, however in exchange it simplifies the data, which improves the performance of the machine learning classifier being used and reduces the chance of overfitting. Given this trade-off, it is important to preserve a certain amount variance, usually above 90%, after reducing the dimensionality. This preserved variance is called explained variance.

The use of PCA does not always prove to be beneficial, yet in the field of baby cry classification, studies such as Sahak et al. (2010b), Sharma et al. (2015), where a Support Vector Machine classifier was implemented with and without PCA to detect asphyxiated infant cry, feeding it features extracted from the Baby Chillanto database, have concluded that this method can raise accuracy in classification, achieving results of 93.84% without PCA and 94.17% with PCA.

4.2 Cry Pattern Classification

In this section are presented the traditional machine learning classifiers that have been mostly used to analyse baby cry. The sole choice of traditional classifiers derives from the lack of a sizeable database to work with, which leads to the incapability of using deep neural networks.

4.2.1 K Nearest Neighbors

The K-Nearest Neighbours (K-NN) classifier is a proximity-based approach where there is a fixed number of k points, in a certain region centred on a feature vector x, known as nearest neighbours of x. This is a supervised learning algorithm, meaning it is trained by using labelled data, which should not be a problem in this work since the database used have been carefully labelled.

First the dataset is divided into a training and testing dataset, where for classification purposes, the label of the latter is unknown. The training phase consists of assigning feature vectors, in this case audio characteristics that represent a cry, to class labels to build the training set. Afterwards the testing dataset is classified and represented as a vector in the feature space. Finally, by resorting to a distance metric, the distances from the test vector and all feature vectors are calculated and the new vector is classified to a particular class, depending on the class label that is found in majority among the k neighbours.

This algorithm is often described as a lazy learning algorithm due to its training phase that is nothing more than the storing of a labeled dataset.. The benefit of using a lazy algorithm lies in the fewer computational costs it requires to train when compared to eager algorithms.

4.2.2 Support Vector Machines

Support Vector Machines (SVM) is the most popular approach used to classify infant cry and in general is a fairly used pattern recognition technique for achieving analogous and sometimes superior results when compared with Artificial Neural Networks. SVM are based on statistical learning theory, developed by Vapnik (1995). To first explain how SVM work, the concept of hyperplane must be defined. Hyperplane classifiers are the class of functions that SVM classifiers are based on (Hearst et al., 1998), so, it classifies data by finding the most optimal hyperplane, meaning the one with the maximal margin between the training data and the decision boundary of two classes. The hyperplane definition recurs to solving a constrained quadratic optimization problem, which raises practicality since it only has one solution. Not all classification problems have a simple hyperplane, therefore in order to solve those problems an additional separating criterion must be used. Kernels are transformation functions that, although they are not exclusive to, are mainly used in SVM to handle non-linear separation problems. Examples of kernel functions are Gaussian Radial Basis Function, polynomial and exponential kernels. The selection of a kernel can significantly alter the architecture of a learning machine. In its basic form,

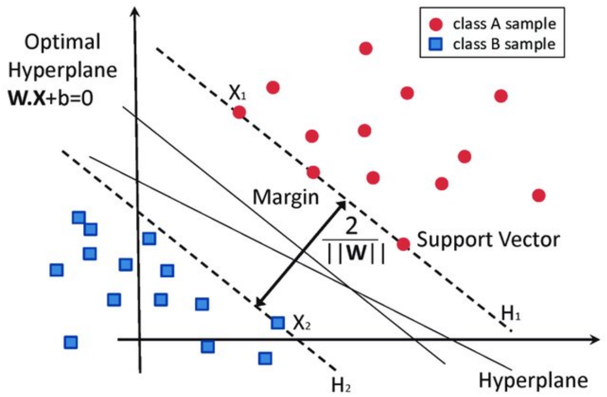


Figure 4.3: Classification of data by support vector machine (SVM) (García-Gonzalo et al., 2016)

SVM are supposed to only support binary classification, however it is possible to apply this classifier to a multiclass problem by breaking it down into multiple binary classification problems. A one-vs-one approach is when a multiclass problem is divided into one binary classification problem for each pair of classes, and a one-to-rest or one-vs-all approach is when a multiclass problem is split into one binary problem for each class.

4.2.3 Random Forest

The Random Forest (RF) algorithm gets its name due to being a collection of multiple random decision trees, which are binary trees that split a dataset until the path from the root node to the leaves only shows at least one unique classification outcome for each existing class. This method alone, however, is highly sensible to the training data, which can result in the model being incapable to generalize when tested with samples that were not present in the training data.

RF combats this by first performing what is called bootstrapping. Bootstrapping consists in randomly selecting samples from the original dataset to build new datasets that do not represent its entirety, yet every dataset has the same size of the original, meaning some samples are replicated. Each new dataset will originate a different decision tree. Bootstrapping ensures that the generated trees are dissimilar from each other, given that they receive different samples from the original dataset, creating a classification model that is less sensitive to the training data.

The next step is to randomly select a subset of features that each tree will use in this training phase only. This number is arbitrary although usually the best results are obtained when using values such as the logarithm or the square root of the total number of features. This randomly selected subset provides two advantages in classification, the first one being that the correlation between the trees that are generated is lowered, and, given that not all features are being used, the trees will not generate similar decision nodes, therefore they can achieve different results. The final step of this algorithm is called aggregation, where a new sample is fed into the forest and each tree predicts its outcome. This new sample is classified as the most predicted label from the decision trees. In addition, the combination of bootstrapping and aggregation is called bagging. The interest in this algorithm comes from the fact that the samples in the dataset that is being used are not linearly separable and although SVM also provide a solution to these non-linear problems, this algorithm offers another approach to solve them.

4.2.4 Minimum Distance Classifier

The Minimum Distance Classifier (MDC) is an algorithm that uses distance functions to measure the dissimilarity between samples in order to classify them. Fairly used in pattern recognition, more specifically in image recognition, the MDC is known for its simplicity and swiftness when compared to other classifiers such as the ones stated above. These characteristics accrue from the process used to classify samples, which consists in calculating the value of a distance function between the sample that is to be classified and all the samples that are already labelled, returning the class that the sample is most similar to. The distance functions used to measure this similarity are usually either the Euclidean distance function or the Mahalanobis distance function. This swiftness and simplicity can sometimes come with a foreseeable trade-off, that is a lower accuracy percentage, however, given the initial idea for this work, to have an application performing real-time classifications, it is pertinent to test the performance of quicker algorithms, since that although other traditional machine learning algorithms and deep learning algorithms are known to achieve better results, they come with a much higher time cost.

4.3 Experimentations & Discussion

After starting the application distribution process, as stated in the intermediate report, it was given continuation to the experimental work. It was chosen to build the algorithms in Python, due to its useful libraries, the main ones important to point out being the sklearn library, which was used to implement the traditional machine learning algorithms, and the librosa library, which was used to extract all the audio features from the recordings.

In the preliminary experiments, the dataset being used were 1 second clips of each audio pain cries and hunger cries from the Baby Chillanto database, fed into an SVM classifier. A problem spotted with this approach was that since the sample and target array fed into the classifier was comprised of one second clips of larger duration audios, when performing the stratified k-fold method, due to the randomization of the samples, it was not possible to infer the accuracy of the prediction of the larger audio. For example, a cry with 5 seconds of duration will generate 5 samples that will be predicted individually, making it impractical to account if the majority of the samples was correctly classified. In order to be able to infer if the full clip of audio is classified correctly, the full clips were used instead of the one second clips, with the downside of diminishing the sample number to 32 samples of hunger cries and 27 samples of pain cries, for a total of 59 samples, with durations ranging from 3 to 20 seconds. Another problem with the classification process in the first semester was that no normalization or scaling was applied to the samples. In this new experimental work, scaling is applied to the train samples and the test samples using the train samples mean and standard deviation.

4.3.1 Experiment 1: Feature Extraction (Baby Chillanto)

Regarding the features used in the first semester, although experiments were conducted not only with MFCC but with other features, namely root mean square, spectral flatness, centroid, bandwidth and roll off, there was no feature selection nor feature reduction performed. It was also noted on the intermediate report that the fundamental frequency was a very common and resourceful feature when analysing baby cry. With that in mind, the feature extraction process started by extracting the fundamental frequency of each recording in addition to the previously mentioned features, adding up to a total of 882 features.

The next step was to implement other classifiers, specifically K-NN, RF and MDC, with the purpose of comparing their performance when classifying the dataset samples. As it was performed for the SVM, which had cost and gamma values ranging from 2^{-13} to 2^{13} and 2^{-13} to 2^{0} respectively, each new classifier had parameters that had to be inferred. The K-NN algorithm, selected due to its simplicity, being an unsupervised learning algorithm, had various values of K ranging from 1 to 20, in order to find a suitable value where the predictions were stable yet the number of errors was minimal. In the RF algorithm, given that there were three parameters being tested, those being the number of estimators, also known as number of decision trees, maximum depth of each tree and maximum number of features selected for each tree, some preliminary attempts were run where time was taken into account. For the number of estimators, given that values ranging from 50 to 1000 were used, 50 and 100 were the number of estimators that allowed for a better classification; for the maximum depth of the trees, from the values chosen ranging from 10 to 100, 10 proved to be sufficient to construct all the decision trees; at last for the maximum number of features to select for each tree, the options of either using the square root or the base 2 logarithm of the total number of features were tested, resulting in the conclusion that in this case most of the classifications achieved higher accuracies when using the square root of the total number of features. In the MDC, the only parameter that could be chosen was the distance function, which could be one of two options: Euclidean distance or Mahalanobis distance, where it was observed that the Mahalannobis distance function allowed for better results.

After inferring the parameters for each classifier, this new experiment commenced by analysing how the newly implemented classifiers performed when fed the MFCC features alone and the MFCC features combined with the other extracted audio characteristics mentioned above, without any kind of feature selection or reduction. Due to the already mentioned variation of the sample rate of the audio files in Chapter 2, the files were re-sampled to 8kHz.The feature extraction process was done differently twice. In the first extraction the focus was on solely obtaining MFCC, since they are known as baseline features in audio feature extraction. By specifying a length of the FFT window of 1024, with a window length of approximately 743 where the rest of the window is padded with zeros to match the FFT window, a hop length of approximately 186, 40 coefficients were extracted for each audio file. Afterwards, statistical modelling was applied to the feature vector, representing now seven statistics: mean, standard deviation, skewness, kurtosis, median, maximum and minimum, obtaining a matrix with the dimensions of 40 by 7, which was flattened. Finally, the feature matrix is composed of all the audio files mentioned where each has a feature vector with the dimensions of 280. Then for the second feature set, in addition to the extraction of MFCC, the other frequency, timbre and intensity features mentioned were also extracted and statistical modelling was also applied, generating a total of 882 features.

Since the number of samples of this experiment was significantly low, it was noted that it was better to perform a cross-validation of 5 folds instead of the 10 folds. This cross-validation was performed 20 times to generate a total of 100 different models. The results seen in Table 4.1 showcase the average metrics from the predictions made by those 100 models.

Table 4.1: Experiment 1: MFCC features (280) vs MFCC features + Extra (882) results

Classifier	**Features**	**Accuracy**	**F1-Score (Pain)**	**F1-Score (Hunger)**	**Recall (Pain)**	**Recall (Hunger)**	**Precision (Pain)**	**Precision (Hunger)**
KNN	MFCC	76.60%±9.14%	**70.46%±14.11%**	79.73%±8.67%	69.00%±19.05%	82.21%±15.31%	76.94%±17.93%	79.87%±10.06%
SVM (Linear)	MFCC	73.18%±8.85%	54.19%±25.43%	80.21%±5.47%	46.00%±26.46%	93.07%±9.78%	77.58%±31.6%	71.80%±9.55%
SVM (Polynomial)	MFCC	75.05%±10.44%	54.92%±27.10%	**82.33%±6.43%**	43.4%±24.99%	98.21%±4.85%	83.48%±33.68%	71.44%±9.25%
SVM (RBF)	MFCC	71.72%±10.03%	47.73%±28.37%	80.10%±5.93%	37.80%±25.44%	96.78%±6.70%	73.42%±39.46%	69.12%±9.09%
Random Forest	MFCC	71.92%±11.47%	62.19%±16.78%	77.05%±9.80%	57.20%±20.00%	82.67%±13.98%	73.42%±19.96%	73.31%±10.61%
MDC	MFCC	67.40%±12.65%	60.29%±15.24%	71.20%±13.11%	59.80%±19.70%	72.95%±18.48%	65.20%±19.09%	71.74%±12.44%
KNN	MFCC + Extra	**77.39%±8.11%**	68.49%±13.65%	81.71%±7.40%	61.20%±20.09%	88.86%±12.81%	84.61%±14.95%	77.49%±9.81%
SVM (Linear)	MFCC + Extra	**75.08%±8.95%**	59.71%±21.93%	81.39%±5.88%	50.60%±23.74%	93.10%±8.48%	82.66%±24.79%	73.35%±9.72%
SVM (Polynomial)	MFCC + Extra	66.62%±7.49%	32.30%±22.63%	77.66%±4.18%	21.8%±17.91%	99.69%±2.17%	74.55%±43.16%	63.88%±6.14%
SVM (RBF)	MFCC + Extra	73.55%±8.31%	55.30%±21.83%	80.78%±5.09%	44.20%±21.41%	95.33%±7.11%	83.77%±27.62%	70.81%±7.94%
Random Forest	MFCC + Extra	73.12%±11.63%	63.04%±17.69%	78.23%±10.00%	57.20%±20.60%	84.86%±14.12%	76.54%±20.07%	73.72%±10.27%
MDC	MFCC + Extra	69.79%±13.22%	59.47%±20.43%	75.19%±10.88%	55.60%±22.55%	80.12%±14.90%	68.02%±23.67%	72.21%±11.65%

Table 4.2: Experiment 2: Feature Selection without PCA (50) vs Feature Selection with PCA (21) results

Classifier	**Pre-processing**	**Accuracy**	**F1-Score (Pain)**	**F1-Score (Hunger)**	**Recall (Pain)**	**Recall (Hunger)**	**Precision (Pain)**	**Precision (Hunger)**
KNN	Feature Selection, no PCA	76.62%±9.69%	70.75%±13.96%	79.81%±8.80%	69.80%±19.29%	81.69%±14.10%	76.46%±16.02%	79.94%±10.44%
SVM (Linear)	Feature Selection, no PCA	76.55%±8.91%	64.59%±21.13%	81.45%±6.53%	59.60%±26.68%	89.10%±12.96%	82.15%±22.28%	77.44%±11.61%
SVM (Polynomial)	Feature Selection, no PCA	65.84%±6.69%	31.08%±22.48%	77.02%±3.69%	21.20%±17.16%	98.69%±4.17%	69.08%±44.04%	63.40%±5.12%
SVM (RBF)	Feature Selection, no PCA	**78.08%±8.81%**	68.97%±16.68%	**82.39%±6.93%**	63.40%±22.28%	89.10%±11.98%	84.32%±17.09%	78.40%±10.60%
Random Forest	Feature Selection, no PCA	69.45%±11.55%	59.72%±17.05%	74.54%±10.56%	56.20%±20.53%	79.31%±15.44%	68.74%±20.28%	72.06%±11.40%
MDC	Feature Selection, no PCA	68.89%±12.13%	61.26%±16.21%	73.22%±11.08%	60.20%±19.90%	75.24%±15.16%	66.30%±18.24%	72.95%±11.74%
KNN	Feature Selection with PCA	**78.03%±11.03%**	**73.34%±14.41%**	80.39%±10.86%	73.80%±20.14%	81.17%±16.82%	77.77%±17.15%	82.68%±12.06%
SVM (Linear)	Feature Selection with PCA	**78.07%±10.43%**	67.43%±22.04%	**82.67%±8.03%**	61.20%±23.80%	90.48%±11.97%	80.48%±25.38%	77.48%±10.20%
SVM (Polynomial)	Feature Selection with PCA	64.27%±6.51%	24.92%±21.81%	76.37%±3.55%	16.40%±16.09%	99.55%±2.58%	61.13%±48.12%	62.15%±5.14%
SVM (RBF)	Feature Selection with PCA	75.03%±8.73%	59.25%±21.39%	81.41%±5.84%	49.60%±23.58%	93.83%±9.02%	85.52%±23.59%	72.99%±9.29%
Random Forest	Feature Selection with PCA	67.23%±10.92%	50.82%±22.37%	74.32%±9.04%	45.40%±24.31%	83.50%±15.04%	66.75%±29.17%	68.69%±9.85%
MDC	Feature Selection with PCA	70.32%±11.41%	66.45%±12.58%	71.37%±15.43%	70.80%±19.88%	70.02%±21.28%	67.25%±16.24%	77.48%±14.89%

Regarding the use of the full duration of each audio as a sample, in contrast to what was performed in the first semester where each sample represented the data of each second of the recordings, it can be noticed that in general the SVM classifier obtained slightly better results in terms accuracy, along with lower values of standard deviation. This improvement can be attributed to the fact that each sample now has more information, leading to a more discernible classification.

Aside from the SVM classifier with the polynomial kernel, it was observed that all classifiers gained a slight boost in accuracy, however this can arguably be called an improvement, given that for that increase to happen it was necessary to feed almost 4 times the number of features into the classifiers for a trade-off of not more than 2% of accuracy and a slighter decrease of the standard deviation. Nevertheless, it is clear that it may be beneficial to use more than the MFCC to predict a recording, so it was of interest to see how that feature set would influence the classifiers once it passed through feature selection and possibly feature reduction.

In terms of accuracy, the best result was achieved by the K-NN classifier with 882 features, with an accuracy of 77.39% and a standard deviation of 8.11%, yet it is still important to analyse the F1-scores from each class. In general, the F1-score of the pain class is significantly lower and the standard deviation is also considerably higher than the F1-score of the hunger class and respective standard deviation, which means that the pain cry classification is severely hindering the overall classification accuracy. Examining further, given that the F1-score is a weighted average of precision and recall, these values were analysed and it was noticeable that, although both classes have similar average values of precision, the pain cries had a much bigger oscillation of values. As for the recall, the pain class had an overall very poor average, some being below 50%, therefore influencing the F1-score negatively. This means that, although the number of false positives in the pain class is relatively low, the classifiers are not able to correctly predict the pain cries most of the time, resulting in a low ratio of correctly predicted pain cries to all pain samples.

There may be several factors that contribute to this poor classification, e.g., the fact that there are less pain cries both in number and duration when compared to the number of hunger cries and there may be a problem with certain recordings that lead them to always fail classification. With the purpose of understanding the latter, each misclassification was checked in order to verify if the misclassified audios were always the same, meaning there may be a problem with the recordings themselves, or if it was always random. After verifying it was reckoned that some audios would indeed fail every prediction, which led to listening to said audios to look for reasons that may influence this problem. It was noted that the audios that failed consistently often had other background noises present in the recordings, such as people talking and even other babies crying. In general, this dataset had a certain distortion present in each recording, being even more noticeable in the cries that were misclassified the most. This distortion is not only audible, but it also influenced some characteristics extracted, such as the fundamental frequency. As stated in Section 2.1, a healthy baby has a fundamental frequency that ranges from 200Hz to 600Hz, yet the average fundamental frequency of each audio ranges from 1000Hz to 2205Hz, being that that maximum value is the limit captured by the function responsible for the extraction of this feature. Those values are considered to only be produced by babies with some health impairment and, although the dataset contains samples of baby cries of asphyxia and deafness, the samples used were not labelled as such.

4.3.2 Experiment 2: With and without feature reduction (Baby Chillanto)

With this new feature set achieving better results, a new experiment was created to focus on improving the classification accuracy and swiftness by pre-processing it. Firstly, the feature selection process ensued, starting by removing low variance features, those being any feature that had a variance lower than 0.16. This step removed a total of 324 features, leaving 558 features remaining.

With the low variance features removed, the Pearson Correlation method was applied to the feature vector, where the correlation of the remaining features was calculated and, after obtaining a correlation matrix, if the correlation between two features was higher than 90%, then it was also necessary to infer the correlation between said features and the class. The feature with the lowest correlation to the class was removed from the feature vector. With this method, 58 features were eliminated, leaving the feature set with 500 features. Next, the feature selection algorithm Minimum Redundancy Maximum Relevancy was applied to the remaining features. In order to assess the minimum number of features necessary to achieve the best classification accuracy, several runs of the K-NN classifier were performed, starting the run with the ten features ranking highest and incrementing the testing feature set with the next ten highest ranked features in each iteration, generating an elbow graph of the accuracy results for visualization purposes. The x axis represents the number of features, by the tens, in the feature set and the y axis represents the average percentage accuracy achieved in each model.

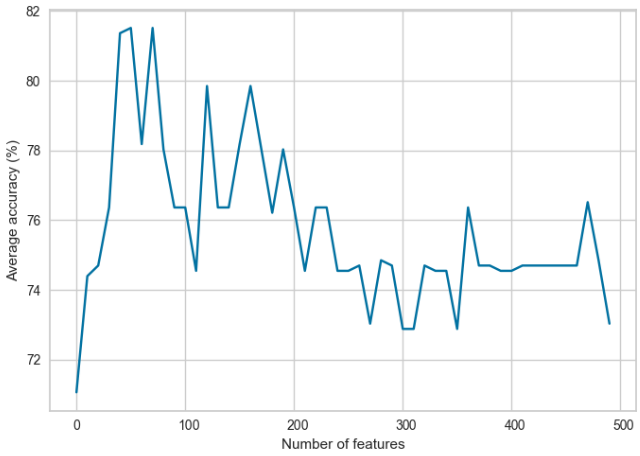


Figure 4.4: Elbow Graph for the first experience

From this graph, labelled as Figure 4.4, the features used for classification were reduced to the 50 highest ranking features from the MRMR algorithm. After this step, a lot of features were removed and some, such as flatness and spectral bandwidth, were removed entirely. Given that MRMR ranks features based on their relevancy combined with their redundancy, it is also worth mentioning that, from the remaining fundamental frequency features, the average F_0 was ranked number one and the skewness was also fairly well ranked. The delta-delta features were also significantly present, unlike the MFCC and the delta features, which can be justified by the fact that the delta-delta from the MFCC provide a better notion on how the MFCC varies throughout the recording, unlike MFCC that just provide information at one particular instant.

In addition, the influence of the PCA method was also tested by running the remaining features from the feature selection techniques mentioned above through the classifier, with and without PCA. After applying PCA, the feature set was reduced to contain only 21 features. In Table 4.2 are displayed the results of this experiment, where it is compared how the classifiers performed when being fed the 50 selected features without PCA and when being fed the reduced 21 features with PCA.

Once again, in terms of accuracy, SVM displayed the best results, the best one being the SVM with the RBF kernel, combined with the feature set that only suffered feature selection, with an accuracy of 78.08% and a standard deviation of 8.81%. The K-NN classifier also showed once again to be able to deliver one of the best results of 78.03% and a standard deviation of 11.03% and the highest F1-score of pain cries of 73.34% when fed the feature set that suffered feature selection and PCA. Even though both classifiers mentioned achieved fairly similar results, there is one aspect that must be taken into account that was not mentioned in the table. This study aimed to look for the most suitable classifier to have in a mobile application, which would have to perform real time classifications, therefore, it is important to have a classifier that is able to perform the fastest. The SVM classifier may take a considerable amount of time in training when compared with the K-NN training time, but the model can be trained beforehand. With a trained model the SVM will outperform the K-NN classifier as more samples are added to the feature set, since K-NN takes into account every feature point to calculate the approximation of new datapoints. The SVM classifier can also handle outliers in a more effective way, since they can be ignored by controlling the cost value, where a low cost value allows for more outliers and a high cost value allows for less outliers.

As for the use of feature reduction techniques, as expected, PCA did not always provide an improvement to the classification accuracy, as it can be seen in the Random Forest, SVM with the RBF kernel, where the accuracy was lowered, yet it is possible to observe that the main difference lied once again on the F1-score of the pain cries, leading to conclude that, when applying the feature reduction in those classifiers, the loss of dimensionality causes the pain cries to become more difficult to discern.

Both the RF and MDC continued to show poor results, barely achieving 70% accuracy, yet, as mentioned in the last experiment, this may not mean that said classifiers are less suitable for baby cry prediction. The results may simply derive from the fact that the sample size was fairly limited, which hindered the classification of correct pain cry prediction the most which was also witnessed in all the F1-scores of this experiment. The hunger cries remained with a good F1-score yet the pain cries, especially the values obtained by the SVM classifier with a polynomial kernel, showed once again to have a very low recall or sensitivity ratio, which influenced the F1-score negatively.

The values obtained in this experiment may not have shown significant increase in terms of accuracy when compared with the previous experiment, however it is important to notice that these results were obtained with a far smaller feature set of 50 features selected from the original 882, and 21 features in the case of the feature set with PCA, meaning that the model complexity of each classifier and training time was greatly reduced yet the prediction outcome remained acceptable.

4.3.3 Experiment 3: With and without feature reduction (Donate-a-Cry)

Even though there may have been some improvement regarding the number of features used, the results obtained still pale in comparison to the other studies with the same classifiers. This may be due to a multitude of reasons, the main one being that in the studies where good results were achieved, the datasets used contained more data, leading to a possible better discerning between classes. It is also possible that the studies did not measure accuracy so meticulously as performed in this work. Nevertheless, given that there was another available dataset to be studied the second experiment was repeated, this time using the public Donate-a-Cry database. The processing of the 882 features was performed once more, this time, after removing low variance features and applying the Pearson correlation method, the MRMR ranked the remaining 556 features and a new elbow graph was generated, as it can be seen in Figure 4.5 from which it was concluded that the feature set should be comprised of the best 220 features. The impact of PCA was also tested using this dataset, reducing the feature set from 220 features to 83 features when applied.

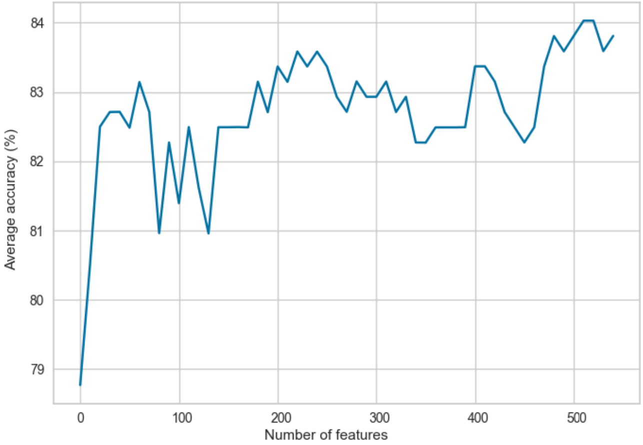


Figure 4.5: Elbow Graph for the second experience

Upon closer inspection of the dataset, it was discovered that it was heavily imbalanced, in the sense that, from the total 457 cries, 382 of them were labelled as hunger cries, meaning that in order for a classifier to have a train set and a test set containing samples from all the 5 classes, 5 fold cross-validation had to be used once more. For this experiment, due to the imbalanced dataset, two more metrics were taken into account, those being the weighted f1-score and the macro f1-score, to show the difference between taking label imbalance into account or not.

As it can be seen in Table 4.3, this imbalance led the classifiers to only be able to discern the cries of hunger successfully, since classifiers tend to pick the majority class when facing an imbalanced problem, leading to an overall good accuracy percentage, yet all the other types of cry were misclassified heavily, many times having an f1-score of 0%. This can be noticed most frequently in the f1-scores of the eructation cries, from which only the MDC with a feature set without PCA and the K-NN with a feature set with PCA were able to discern any cries from that type, due to it being the cry with the lowest amount of samples, having only 8 recordings. The MDC in specific had interesting results, since its average classification accuracy was very low and dissatisfactory, yet it was able to achieve the best f1-scores in tiredness and discomfort cries. The values obtained in the macro f1-scores and weighted f1-scores also prove that even this metric can be misleading. In this case, since weighted f1-scores take into account label imbalance, the results obtained portray a much more successful model, when in reality, if the macro f1-score is calculated, where the number of samples per label is not a weighing factor, it will reflect that the models are practically only predicting the majority class correctly.

This experiment helps to confirm that accuracy alone may not always be a reliable metric to measure the performance of a trained model, since that in cases such as this where there is an imbalanced dataset, high accuracy may not mean good performance and a combination of other metrics such as f1-score, recall and precision may present a more accurate measurement. The information of this experiment also leads to believe that the studies that used this dataset, such as Sharma et al. (2015), may have indeed obtained similar classification accuracies using RF, K-NN and SVM classifiers, however, due to the omission or lack of calculation of the f1-scores, it is possible that those classifiers did not have a very good performance when classifying cries that were not hunger cries. Another possible problem is that, as stated in chapter 2, this database is open to submissions from the public, meaning anyone can submit and classify them, which can lead to the mislabelling of some samples, affecting the efficiency of the classifiers.

Table 4.3: Experiment 3: Feature Selection without PCA (220) vs Feature Selection with PCA (83) results

Classifier	**Feature Reduction**	**Accuracy**	**Balanced Accuracy**	**F1-Score (Macro)**	**F1-score (Weighted)**	**F1-Score (Belly Pain)**	**F1-Score (Eructation)**	**F1-Score (Discomfort)**	**F1-Score (Hunger)**	**F1-Score (Tiredness)**
KNN	None	84.45%±0.95%	26.64%±4.29%	27.77%±5.89%	78.35%±1.64%	38.46%±22.09%	0%±0%	7.60%±14.16%	91.48%±0.55%	1.28%±6.31%
SVM (Linear)	None	83.74%±0.80%	20.79%±2.19%	19.41%±3.13%	76.51%±1.32%	1.17%±8.25%	0%±0%	1.86%±7.37%	91.17%±0.49%	2.84%±9.12%
SVM (Polynomial)	None	83.59%±0.64%	20.00%±5.55%	18.21%±0.08%	76.12%±0.90%	0%±0%	0%±0%	0%±0%	91.06%±0.38%	0%±0%
SVM (RBF)	None	83.59%±0.64%	20.00%±5.55%	18.21%±0.08%	76.12%±0.90%	0%±0%	0%±0%	0%±0%	91.06%±0.38%	0%±0%
Random Forest	None	83.59%±0.64%	20.00%±5.55%	18.21%±0.08%	76.12%±0.90%	0%±0%	0%±0%	0%±0%	91.06%±0.38%	0%±0%
MDC	None	37.22%±5.55%	27.51%±7.70%	18.61%±3.58%	46.94%±5.35%	15.68%±8.13%	0.15%±1.53%	11.75%±7.64%	53.95%±6.15%	11.51%±10.30%
KNN	PCA	84.27%±0.93%	25.63%±5.01%	26.29%±6.78%	77.99%±1.72%	31.01%±24.44%	0.67%±6.63%	6.50%±13.99%	91.37%±0.55%	1.90%±7.56%
SVM (Linear)	PCA	83.69%±0.70%	20.36%±1.25%	18.79%±2.02%	76.34%±1.14%	0%±0%	0%±0%	1.24%±6.08%	91.14%±0.47%	1.57%±6.87%
SVM (Polynomial)	PCA	83.66%±0.67%	20.20%±1.14%	18.52%±1.73%	76.18%±1.00%	1.50%±8.53%	0%±0%	0%±0%	91.08%±0.40%	0%±0%
SVM (RBF)	PCA	83.59%±0.64%	20.00%±5.55%	18.21%±0.08%	76.12%±0.90%	0%±0%	0%±0%	0%±0%	91.06%±0.38%	0%±0%
Random Forest	PCA	83.60%±0.63%	20.00%±5.55%	18.21%±0.08%	76.12%±0.90%	0%±0%	0%±0%	0%±0%	91.06%±0.38%	0%±0%
MDC	PCA	83.58%±0.65%	20.00%±0.03%	18.21%±0.08%	76.11%±0.90%	0%±0%	0%±0%	0%±0%	91.05%±0.38%	0%±0%

4.3.4 Experiment 4: Testing Baby Chillanto trained models with Donate-a-Cry database

A final experiment was performed with the intent of analysing how a model trained with the Baby Chillanto dataset would perform when classifying cries from the Donate-a-Cry dataset. Due to the already mentioned imbalance of the Donate-a-Cry database and the fact that only the cries of hunger and pain from the Baby Chillanto database were used, the cries from the former were processed as a binary classification problem. This means that, from the Donate-a-Cry database, the feature set was processed to have cries of hunger and all the other types of cry were labelled equally as “other”, making the purpose of this experiment to test how well a model trained with the Baby Chillanto dataset would predict cries of hunger.

This experiment compared once more the results of using only feature selection, having a feature set of 50 features, and using feature selection and PCA, having a feature set of 21 features. Each model was trained and tested once for each variable parameter specific to each classifier as it was done for the previous experiments and the best outcomes were documented. The results are expressed in Table 4.4.

Table 4.4: Experiment 4: Testing Baby Chillanto trained models with Donate-a-Cry database results

Classifier	**Pre-processing**	**Best Parameters**	**Accuracy**	**F1-Score (Hunger)**	**F1-Score (Other)**
KNN	Feature selection, no PCA	K: 20	**78.77%**	**87.83%**	**17.09%**
SVM (Linear)	Feature selection, no PCA	Cost: 2^{-13} Gamma: 2^{-13}	83.59%	91.06%	0%
SVM (Polynomial)	Feature selection, no PCA	Cost: 2^{-13} Gamma: 2^-2	83.81%	91.17%	2.63%
SVM (RBF)	Feature selection, no PCA	Cost: 2^{-13} Gamma: 2^{-13}	83.59%	91.06%	0%
Random Forest	Feature selection, no PCA	Estimators: 50 Max depth: 10 Max features: Sqrt(50)	75.05%	85.23%	19.72%
MDC	Feature selection, no PCA	Algorithm: Mahalanobis	77.68%	87.09%	17.74%
KNN	Feature selection, with PCA	K: 9	78.77%	87.80%	18.49%
SVM (Linear)	Feature selection, with PCA	Cost: 2^{-13} Gamma: 2^{-13}	83.59%	91.06%	0%
SVM (Polynomial)	Feature selection, with PCA	Cost: 2^{-13} Gamma: 2^-2	83.81%	91.17%	2.63%
SVM (RBF)	Feature selection, with PCA	Cost: 2^{-13} Gamma: 2^{-13}	83.59%	91.06%	0%
Random Forest	Feature selection, with PCA	Estimators: 1000 Max depth: 10 Max features: Sqrt(21)	65.86%	77.78%	**26.42%**
MDC	Feature selection, with PCA	Algorithm: Mahalanobis	81.84%	89.96%	4.60%

In the cases where the f1-score is of the “other” cries was 0%, once again it can be seen that the accuracy may be high yet this means very little when it is known that the classifier failed to discern a single cry from that type. Since it is a binary problem, this shows that the best prediction achieved by these models was predicting that every sample belonged to the majority class, which is far from ideal. On the other hand, there is the case of the RF classifier with feature selection and PCA, that achieved the lowest classification accuracy of 65.86%, yet it was the classifier that discerned the most amount of cry samples that were not hunger cries.

Overall, apart from the models that were only able to predict hunger cry, these results are somewhat satisfactory since there are clear audible differences in the files from each database, yet the models still managed to discern hunger cries from other types of cry.

Taking into account the results obtained in all the four experiments, it can be said that the best option to implement in a real time baby cry prediction mobile application would be using SVM, however it can also be noted just because some classifiers performed poorly, it does not mean they would not be fit for the purpose. For example, the Random Forest classifier may not have achieved the most satisfactory results, yet it could have been a different scenario if there was a larger database to work with.

Chapter 5

Conclusions and Future Work

The decoding of baby cry is a challenge as old as humanity and it can be said that in the last decades the progress achieved in the field of pattern recognition has opened many doors that allow to understand other perspectives on how to approach this subject by resorting to the computational analysis of audio samples. This thesis focused on contributing to that progress by implementing and testing multiple baby cry classifiers with the available data and by developing a mobile application that can be a useful tool to gather audio recordings of baby cry and form a database, since gathering baby cry is relatively hard due to ethical reasons and privacy issues. To achieve these goals, the background study of baby cry related concepts in Section 2.1 combined with the analysis of previously performed experiments described in Chapter 2, deeply influenced the direction of this work, since it gave a notion of which audio characteristics play a major role when it comes to discerning baby cry and what has been previously attempted, regarding machine learning models.

The studies described were selected due to their recurring use and satisfactory results, leading to the selection of Support Vector Machines, K-Nearest Neighbours, Minimum Distance Classifier, Random Forest classifiers for the comparative experiments performed in this work. It is known that the results obtained could have been greater with a larger dataset, however, with the available audio samples it can be considered that a classifier such as K-NN can achieve fairly good results in terms of prediction accuracy, yet it has to be kept in mind that this is a lazy algorithm that does not perform that well, having a higher classifying time cost the larger the number of samples is. Given that the original idea was to have a classifier able to work swiftly in a real time baby cry decoding mobile application, an SVM can be considered the best option out of the four in the study, if the right kernel, cost and gamma value is selected. This does not mean that ultimately this classifier is superior to the others. As it was mentioned in Section 4.2, every classifier has potential drawbacks and strengths that depend on the number of samples, features extracted and specific parameters. Regarding the data collection, even though it was not possible to gather data as expected, the work can still be considered positive, since the app can still be used for future endeavours and allow for the use of different approaches, which leads to the next and closing section.

Regarding future work, as mentioned above, the lack of data shifted the direction of this work. With more audio samples it is possible to approach this problem from other angles, the main one being resorting to Deep Learning approaches, which were briefly discussed and considered in Chapter 2. A classifier such as Convolutional Neural Network can be useful in the sense that the data does not need to be pre-processed, since the neural network itself extracts features.

The use of the recording application may also be considered for a continuation of this work to gather data and, if successful, a reliable classifier can be integrated into it. With time and data, the application could also have an option to retrain the model and its parameters with newly gathered samples from the users, in order to attain a classifier specially adapted to their baby.

References

Abbs, K. J. (2015). Dysphonations in infant cry: A potential marker for health status (Doctoral dissertation, Bowling Green State University).

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., ... & Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. Electronics, 8(3), 292.

Asthana, S., Varma, N., & Mittal, V. K. (2014, December). Preliminary analysis of causes of infant cry. In 2014 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT) (pp. 000468-000473). IEEE.

Asthana, S., Varma, N., & Mittal, V. K. (2015, February). An investigation into classification of infant cries using modified signal processing methods. In 2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 679-684). IEEE.

Baeck, H. E., & Souza, M. N. (2001, October). Study of acoustic features of newborn cries that correlate with the context. In 2001 Conference Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (Vol. 3, pp. 2174-2177). IEEE.

Balandong, R. P. (2013). Acoustic Analysis of Baby Cry. Biomedical Eng. University of Malaya.

Bano, S., & RaviKumar, K. M. (2015, February). Decoding baby talk: A novel approach for normal infant cry signal classification. In 2015 International Conference on Soft-Computing and Networks Security (ICSNS) (pp. 1-4). IEEE.

Barr, R. G., Fairbrother, N., Pauwels, J., Green, J., Chen, M., & Brant, R. (2014). Maternal frustration, emotional and behavioural responses to prolonged infant crying. Infant Behavior and Development, 37(4), 652-664.

Bhagatpatil, V. V., & Sardar, V. M. (2014). An automatic infant's cry detection using linear frequency cepstrum coefficients (lfcc). International Journal of Scientific & Engineering Research, 5(5), 1379-1383.

Bradbury, J. (2000). Linear predictive coding. Mc G. Hill.

Bănică, I. A., Cucu, H., Buzo, A., Burileanu, D., & Burileanu, C. (2016, June). Automatic methods for infant cry classification. In 2016 International Conference on Communications (COMM) (pp. 51-54). IEEE.

Chaiwachiragompol, A., & Suwannata, N. (2016). The features extraction of infants cries by using discrete wavelet transform techniques. Procedia Computer Science, 86, 285-288.

Cohen, R., & Lavner, Y. (2012, November). Infant cry analysis and detection. In 2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel (pp. 1-5). IEEE.

Daga, R. P., & Panditrao, A. M. (2011). Acoustical analysis of pain cries in neonates: Fundamental frequency. Int. J. Comput. Appl. Spec. Issue Electron. Inf. Commun. Eng ICEICE, 3, 18-21.

Davis, S., & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. IEEE transactions on acoustics, speech, and signal processing, 28(4), 357-366. b

Dewi, S. P., Prasasti, A. L., & Irawan, B. (2019, July). The Study of Baby Crying Analysis Using MFCC and LFCC in Different Classification Methods. In 2019 IEEE International Conference on Signals and Systems (ICSigSys) (pp. 18-23). IEEE.

Ding, C., & Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. Journal of bioinformatics and computational biology, 3(02), 185-205.

Dunstan, P. (2012). Calm the Crying Deluxe. Penguin.

Fort, A., Ismaelli, A., Manfredi, C., & Bruscaglioni, P. (1996). Parametric and non-parametric estimation of speech formants: application to infant cry. Medical engineering & physics, 18(8), 677-691.

Fort, A., & Manfredi, C. (1998). Acoustic analysis of newborn infant cry signals. Medical engineering & physics, 20(6), 432-442.

Franti, E., Ispas, I., & Dascalu, M. (2018, July). Testing the universal baby language hypothesis-automatic infant speech recognition with cnns. In 2018 41st International Conference on Telecommunications and Signal Processing (TSP) (pp. 1-4). IEEE.

García-Gonzalo, E., Fernández-Muñiz, Z., García Nieto, P. J., Bernardo Sánchez, A., & Menéndez Fernández, M. (2016). Hard-rock stability analysis for span design in entry-type excavations with learning classifiers. Materials, 9(7), 531.

Gerhard, D. (2003). Pitch extraction and fundamental frequency: History and current techniques (pp. 0-22). Regina, Canada: Department of Computer Science, University of Regina.

Giannakopoulos, T., & Pikrakis, A. (2014). Introduction to Audio Analysis: a MATLAB® approach. Academic Press.

Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., & Scholkopf, B. (1998). Support vector machines. IEEE Intelligent Systems and their applications, 13(4), 18-28.

Ji, C., Mudiyanselage, T. B., Gao, Y., & Pan, Y. (2021). A review of infant cry analysis and classification. Eurasip Journal on Audio, Speech, and Music Processing, 2021(1), 1-17.

Kulkarni, P., Umarani, S., Diwan, V., Korde, V., & Rege, P. P. (2021, April). Child Cry Classification-An Analysis of Features and Models. In 2021 6th International Conference for Convergence in Technology (I2CT) (pp. 1-7). IEEE.

LaGasse, L. L., Neal, A. R., & Lester, B. M. (2005). Assessment of infant cry: acoustic cry analysis and parental perception. Mental retardation and developmental disabilities research reviews, 11(1), 83-93.

Liu, L., Li, Y., & Kuo, K. (2018, March). Infant cry signal detection, pattern extraction and recognition. In 2018 International Conference on Information and Computer Technologies (ICICT) (pp. 159-163). IEEE.

Maghfira, T. N., Basaruddin, T., & Krisnadhi, A. (2020, April). Infant cry classification using cnn–rnn. In Journal of Physics: Conference Series (Vol. 1528, No. 1, p. 012019). IOP Publishing.

Manikanta, K., Soman, K. P., & Manikandan, M. S. (2019, December). Deep Learning Based Effective Baby Crying Recognition Method under Indoor Background Sound Environments. In 2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS) (Vol. 4, pp. 1-6). IEEE.

Mittal, V. K. (2016a, September). Discriminating the infant cry sounds due to pain vs. discomfort towards assisted clinical diagnosis. In 7th Workshop on Speech and Language Processing for Assistive Technologies, SLPAT 2016 (Vol. 2016, pp. 37-42).

Mittal, V. K. (2016b, October). Discriminating features of infant cry acoustic signal for automated detection of cause of crying. In 2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP) (pp. 1-5). IEEE.

Mohapatra, R. K. (2016). Handwritten Character Recognition of a Vernacular Language: The Odia Script (Doctoral dissertation).

Mrazova, I., & Kukacka, M. (2012). Can deep neural networks discover meaningful pattern features?. Procedia Computer Science, 12, 194-199.

Onu, C. C., Udeogu, I., Ndiomu, E., Kengni, U., Precup, D., Sant'Anna, G. M., ... & Opara, P. (2017). Ubenwa: Cry-based diagnosis of birth asphyxia. arXiv preprint arXiv:1711.06405.

Orlandi, S., Garcia, C. A. R., Bandini, A., Donzelli, G., & Manfredi, C. (2016). Application of pattern recognition techniques to the classification of full-term and preterm infant cry. Journal of Voice, 30(6), 656-663.

Parlak, C., Diri, B., & Gürgen, F. (2014, September). A cross-corpus experiment in speech emotion recognition. In SLAM@ INTERSPEECH (pp. 58-61).

Raina P. Daga, & Anagha M. Panditrao (2011). Article: Acoustical Analysis of Pain Cries' in Neonates: Fundamental Frequency. IJCA Special Issue on Electronics, Information and Communication Engineering, ICEICE(3), 18-21.

Reyes-Galaviz, O. F., & Reyes-Garcia, C. A. (2004). A system for the processing of infant cry to recognize pathologies in recently born babies with neural networks. In 9th Conference Speech and Computer.

Rosales-Pérez, A., Reyes-García, C. A., Gonzalez, J. A., Reyes-Galaviz, O. F., Escalante, H. J., & Orlandi, S. (2015). Classifying infant cry patterns by the Genetic Selection of a Fuzzy Model. Biomedical Signal Processing and Control, 17, 38-46.

Rosenblatt, F. (1957). The perceptron, a perceiving and recognizing automaton Project Para. Cornell Aeronautical Laboratory.

Rusu, M. S., Diaconescu, Ş. S., Sardescu, G., & Brătilă, E. (2015, October). Database and system design for data collection of crying related to infant's needs and diseases. In 2015 International Conference on Speech Technology and Human-Computer Dialogue (SpeD) (pp. 1-6). IEEE.

Sahak, R., Mansor, W., Lee, Y. K., Yassin, A. I. M., & Zabidi, A. (2010a, August). Performance of combined support vector machine and principal component analysis in recognizing infant cry with asphyxia. In 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology (pp. 6292-6295). IEEE.

Sahak, R., Lee, Y. K., Mansor, W., Yassin, A. I. M., & Zabidi, A. (2010b). Detection of asphyxiated infant cry using support vector machine integrated with principal component analysis. In 2010 IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES) (pp. 485-488). IEEE.

Sharma, S., Asthana, S., & Mittal, V. K. (2015, December). A database of infant cry sounds to study the likely cause of cry. In Proceedings of the 12th International Conference on Natural Language Processing (pp. 112-117).

Shiruru, Kuldeep. (2015). Neural Network Approach for Processing Substation Alarms. International Journals of Power Electronics Controllers and Converters. 1. 21-28.

Silva, G. V. I. S., & Wickramasinghe, D. S. (2017). Infant cry detection system with automatic soothing and video monitoring functions.

Stevens, S. S., Volkmann, J., & Newman, E. B. (1937). A scale for the measurement of the psychological magnitude pitch. The journal of the acoustical society of america, 8(3), 185-190.

Tejaswini, S., Sriraam, N., & Pradeep, G. C. M. (2016, October). Recognition of infant cries using wavelet derived mel frequency feature with SVM classification. In 2016 International Conference on Circuits, Controls, Communications and Computing (I4C) (pp. 1-4). IEEE.

Tuduce, R. I., Cucu, H., & Burileanu, C. (2018, July). Why Is My Baby Crying? An In-Depth Analysis of Paralinguistic Features and Classical Machine Learning Algorithms for Baby Cry Classification. In 2018 41st International Conference on Telecommunications and Signal Processing (TSP) (pp. 1-4). IEEE.

Tuduce, R. I., Rusu, M. S., Horia, C. U. C. U., & Burileanu, C. (2019, July). Automated baby cry classification on a hospital-acquired baby cry database. In 2019 42nd International Conference on Telecommunications and Signal Processing (TSP) (pp. 343-346). IEEE.

Umesh, S., Cohen, L., & Nelson, D. (1999, March). Fitting the mel scale. In 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258) (Vol. 1, pp. 217-220). IEEE.

Vapnik, V. The nature of statistical learning theory. 1995. NY: Springer.

Xie, Q., Ward, R. K., & Laszlo, C. A. (1993, September). Determining normal infants' level-of-distress from cry sounds. In Proceedings of Canadian Conference on Electrical and Computer Engineering (pp. 1094-1096). IEEE.

Zabidi, A., Khuan, L. Y., Mansor, W., Yassin, I. M., & Sahak, R. (2010, March). Classification of infant cries with asphyxia using multilayer perceptron neural network. In 2010 Second International Conference on Computer Engineering and Applications (Vol. 1, pp. 204-208). IEEE.

Zeskind, P. S., McMurray, M. S., Garber, K. A., Neuspiel, J. M., Cox, E. T., Grewen, K. M., ... & Johns, J. M. (2011). Development of translational methods in spectral analysis of human infant crying and rat pup ultrasonic vocalizations for early neurobehavioral assessment. Frontiers in psychiatry, 2, 56.