THE UNIVERSITY OF NEW SOUTH WALES, AUSTRALIA

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Software Engineering Thesis, 2014

Are You Stressed?
Detecting the onset of stress using mobile phones

Written By: Hariharen VEERAKUMAR (z3258355)

Supervisor: Dr Salil KANHERE
Assessor: Mr Mahbub HASSAN

Submission Date: 28/10/2014

ABSTRACT

This thesis is a detailed procedure and analysis of what is required for a functional solution to detect the onset of stress using a mobile phone. This is a new area of study to which the answer has not been achieved, although similar alternatives have proven more accurate. Through research and development, we have taken a number of key symptoms and processed them using a machine learning algorithm – all within a mobile application: the 'StressTester' . The overall accuracy was most consistently found to be 81.67%. Thus, we have produced a solution that, although does not compare with current state-of-the-art technology, we have tapped into unseen territory regarding this issue that should be treated seriously upon further development of the current solution or other solutions.

DEDICATION

I would like to thank the University of New South Wales as a whole with a special mention to Salil Kanhere. You have helped me on a week-by-week basis to help give me direction in what I aimed to achieve in this thesis. May this thesis also help you in future endeavours to find our solution.

Secondly, I would like to acknowledge the support that the Veerakumar family has given me for as long as I can remember, let alone with my thesis. Their efforts have furthered me to become a person more than I could have imagined, and I know I could not have written this without them.

To my partner of 7 years, Fiona De Silva, who has stuck by my side through thick and thin: your willing nature to help me through my feats is the reason I don't come close to stressing so much, and I don't know what tertiary education would have been like without you.

Most of all, I dedicate this thesis to all those whom are exposed to distress at unhealthy levels – let there be a solution not only to our problems, but to stress detection and prevention.

"It is only natural to stress in our life –the key is to not let it turn into distress."

- HtM, 2014.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

A “stress response” can be defined as:

“A physiological reaction caused by the perception of aversive or threatening situations” (Carlson, 2007)

Humanity’s problems can always be related to stress. If any one person is stressed, they experience “stress responses” – that is, they are feeling a consequence of some form of burden that has been placed upon them physically or mentally. Stress is inevitable in a person’s life, and the amount that people experience can only be minimized through the choices we make – simply because they do not enjoy the stresses that result from risky decisions as an example.

Nevertheless, no matter who you are, the human body can react badly to the stress which you experience: whether it is directly from the stressor, or the stress response. A common example to relate to is an elderly person suffering a heart attack due to an emotional stress. Whilst the stressor is not actually stressing the person through physical means, a physical stress response is still triggered – and a dangerous one at that. On a larger scale, it has been found that physiological responses have significant adverse long-term effects on our health when dealing with continuous stressors, as opposed to episodic stressors. (Carlson, 2007)

Bearing this in mind, we have a strong reason to identify stress responses and assess the levels of stress we are experiencing at the point in time. In understanding what stress responses we experience, we can then act to minimise them. Currently, whilst there are technologies to determine significant stressors and stress responses in our lives, we cannot say it is to a point that it is readily accessible. Furthermore, people do not make conscious efforts to assess their stress, especially during their times of stress. If people were more aware of what stressors and stress responses were doing to them both mentally and physically, the problems that people experience would be reduced or eliminated. In particular, this would be most effective if people were made aware of dangerous physiological reactions at the time they are experiencing it.

For something as important as what has been discussed, and with the new and ever-changing technological world we live in, the issue can now start to be addressed effectively. There are current technologies that we use on a day-to-day basis that can have a further enhanced capability to detect stress responses and, essentially, assess levels of stress.

One particular technology that fits this category is a mobile smartphone (“phone”). Those who associate with current technology will be well aware that phones are capable of doing more than making a voice call. Phones have evolved from simply making vocal communication to messaging, and now have progressed to completely different user experiences and interactions with touch screens replacing buttons. What people are less aware of is the hardware that is contained within a standard phone. Phones are now equipped with a variety of hardware, from microphones to accelerometers, that provides us with more information than older phones. This allows applications to interact with the physical world we live in and, in particular, the user. The hardware can be utilised to interact with the user and gain information regarding the onset of stress through the user’s stress responses.

Hence, we have a probable solution to the problem. Our inability to analyse our stress levels to a certain degree of accuracy is a valid and significant problem. The knowledge we can gain from a valid solution can result in many benefits: a key benefit being a reduction in dangerous stress responses from conscious effort due to awareness. This path of research will further open up avenues to deal with stress. Ultimately, especially with further technological advancement, we will be able to minimise stress responses from a person, and the knowledge regarding our own stress responses is a good first step and foundation towards the solution.

This paper is a research thesis regarding this problem, where we will develop a complementing solution that we can implement. We will explain the differences between the current state-of-the-art technologies, as well as slightly older applications that were considered the latest technology at the time. Such technologies go from current, all the way back to 2007 – giving us an idea of historical research whilst still being state-of-the-art. It is particularly important to understand what has been attempted. This gives us an appreciation for what has currently been created, further inspires our own ideas, and allows us to innovate further based on these solutions. Lots of current implementations use interesting concepts regarding the various signals that were detected, the collection of data from a variety of instruments, and the processing of the physiological and physical data using interesting algorithms. Such algorithms vary from machine learning algorithms e.g. the use of support vector machines (SVMs), to the use of the Volterra series, which is commonly used in electrical engineering applications. Lastly, we can also identify untapped areas of research that may offer a better solution, such as the use of alternate machine learning algorithms like a random forest decision tree (Flach, 2012).

Thus, we will be able to identify not only key differences between each of the application, but their strengths and weaknesses and how these will inspire ideas for the proposed solution. Furthermore, we will discuss a feasible timed structure to approach and implement the solution, which will include slowly integrating the solution into real-world applications.

From here, we discuss our own work. Firstly, we discuss any further technical research in areas of study concerning machine learning and the use of an API of a mobile phone for a technical standpoint. These are important to understand before we discuss the methodologies in which the application was built. For methodologies, each step is as important as each other in ensuring the optimal solution is developed. We will walk through the justifications for each process, and what values it added to our solution.

We will go through our testing methodologies, which is used to ensure that the application is developed robustly. These are done under a strict methodology, and the results were assessed. From here, we gauged whether the data collected was of an accurate nature, and what would cause these inaccuracies.

Once the application was perfected, the system was evaluated, as will be discussed. This evaluation returned a number of results. An evaluation method was also created so as to measure the accuracy of the application – namely using machine learning metrics. This allows us to draw conclusions from our solution and assess its effectiveness.

2. BACKGROUND

2.1 Literature Review

There are many attempts at implementations that can be noted both using and not using a mobile phone. Such successful implementations include "AutoSense" (Ertin, et al., 2011), "SocioPhone" (Lee, et al., 2013), "MoodScope" (LiKamWa, et al., 2013) and "Mood Meter" (Hernandez, et al., 2012), as well as a methodology to remotely manage hypertension (Logan, et al., 2007). As seen here, a majority of successful implementations to address the larger problem uses external hardware to work with the mobile phone.

In studying these methodologies, we can identify accurate procedures used to determine the onset of stress. We can also note dangerous physiological reactions from these implementations, which address the big picture of a key benefit in identifying the onset of stress. On the other hand, with the restricting aspect of measuring stress using a specific means, there is a hole in the accuracy of these applications. It also does not cater for everybody's personal reactions, but rather defines stress at a certain point and determines the onset of stress according to that definition.

2.1.1 Mobile-application solutions, and data collection methods

Measuring hypertension (high blood pressure) is an extremely important for stress management (Carlson, 2007; Logan, et al., 2007). Hypertension is a prime danger, and has a strong association to stress, hence being one of the leading significant stress responses. As a result, methods in (Logan, et al., 2007) to remotely manage it can prove rather helpful as a key element of stress management as a whole. This particular application also specifically applies to concerns for diabetic patients, however is not restricted to that. Using a blood pressure monitor with an active Bluetooth connection, a mobile phone can interact with the hardware to process the data and send it to the respective physician(s). This particular application, however, does not use the mobile phone for direct methods of input – rather it takes care of everything else.

AutoSense (Ertin, et al., 2011) uses "an unobtrusively wearable wireless sensor suite that can collect continuous measurements" (Ertin, et al., 2011). This external hardware communicates with a phone via an application to collect data regarding stress. The solution "focuses on physiological measures monitoring cardiovascular, respiratory, and thermoregulatory systems." (Ertin, et al., 2011). This allows detection of physiological reactions from the hardware, which is then sent to the phone through a low-frequency radio signal. An algorithm is used to collate the data and, ultimately, provide a judgement of stress.

In a similar manner, studies have been conducted to assess stress using other factors (Lu, et al., 2012). In particular, one application recognises stress using skin conductance, the production of cortisol and pupil diameter, whilst looking at the previously discussed heart rate and blood pressure variables. A binary classification algorithm was used to further increase accuracy – in particular, a linear Support Vector Machine (SVM). Furthermore, in the production, data was also collected manually to detect stress. This produced accurate results, however we must bear in mind it is different to other applications as it uses external hardware.

SocioPhone (Lee, et al., 2013) is an application which monitors face-to-face conversations at the core to investigate social interaction. It is classified as an "interaction-aware application" (Lee, et al., 2013). Whilst the focus is not around stress, human interaction often leads to the revealing of other stress responses in the event the person is stressed (Carlson, 2007; Lee, et al., 2013). The symptoms of different characteristics that the application checks for are also similar to those that can be analysed to detect the onset of stress, namely: "sound signals, online turn segmentation[1] and meta linguistic feature extraction[2]" (Lee, et al., 2013).

MoodScope (LiKamWa, et al., 2013) is a sensor that measures the mental state of a user that does not use physical properties. Rather, the aim of the project is to detect the mood of a user to provide a context as an input for other applications which, in turn, enables "context-aware computing" (LiKamWa, et al., 2013) – something that is highly important in creating a successful application, since it allows the application to act dynamically and enhance human-computer interaction, as opposed to acting statically (Dey & Abowd, 1999). MoodScope does not use physiological signals as most do, but rather uses the activity a user has with their phone to determine what mood they are in. MoodScope further uses customised data sets for each user to gradually improve the accuracy of the application for each user. Whilst accuracy is minimal during initial use, it uses a 2-month training period to increase the accuracy of the application in its inferences, raising the accuracy from 66% to 93%, when analysing 32 participants (LiKamWa, et al., 2013)

[1] A "turn" is a continuous speech segment where a person starts and ends her speech (Lee, et al., 2013)
[2] Meta-linguistic features include characteristics of speech that are not classified as the "language". Examples include pace of speech, and whether the type of speech used is assertive or dominant

MyWalk (How, et al., 2013) is a mobile application that, rather than being based around stress in particular, was developed to help reduce gait asymmetry. Gait asymmetry involves physiological reactions to cause abnormal movements of the human body and limbs, and is very common in the event of a stroke. This is particularly useful to this project for us to understand how sensors are used to detect changes in movement as someone walks. Using the accelerometer, the application was able to determine how the person walks, and measure to what degree they were able to walk "normally" - that is, in a straight line, as per the instructions provided. Making use of all three axes of motion and space, the degrees of asymmetry were calculated on average and the variance between each sampled value could be used to assess recovery progress.

Data collection techniques involving speech were also studied, as this is a key factor that is proportional to the mood of a person. The product, AMMON (Chang, et al., 2011), is a library used to detect stress through speech in mobile phone applications specifically. It was developed in C, and tested on a number of subjects. This is partially inspired by a previous project, SoundSense (Lu, et al., 2009). It uses a linear SVM to predict the nature of the mental health. However, it is completely supervised learning, and so was done completely offline with all data included in the library. Ultimately, the outcome was a success, as it achieved its aim in being as accurate as state-of-the-art technology for the same purpose on a PC.

We can draw a number of parallels between each of the applications, whilst also noting key differences in achieving certain elements of our common goal. We can see from the papers (Ertin, et al., 2011), (Lee, et al., 2013), (LiKamWa, et al., 2013) and (Logan, et al., 2007), phones are used for the collecting and processing of data to generate information regarding human interaction. In particular, (Ertin, et al., 2011), (Lee, et al., 2013) and (Logan, et al., 2007) collect physiological information, with (Ertin, et al., 2011) and (Lee, et al., 2013) specifically using this to calculate one's mood, which we can directly infer one's level of stress. (Logan, et al., 2007), on the other hand, collects physiological data to respond to a dangerous, and very key, stress response. (LiKamWa, et al., 2013) does not utilise physiological reactions, yet still manages to detect stress through other interesting means. (Hernandez, et al., 2012) is very different, since it uses other technologies to collect useful data, analyse the data and produce results based off one parameter to scale what mood the person is in.

The methodology used in (Ertin, et al., 2011) and (Lee, et al., 2013) are one of the most advanced, despite (Ertin, et al., 2011) being a relatively older use of phone applications for this purpose. In (Ertin, et al., 2011), the hardware itself is very intricate – it has the capacity to collate many parameters within a very small device that is “comfortable to wear for long hours in the field” (Ertin, et al., 2011). However, we can agree that not having hardware attached to us is, in itself, a superior advantage. (Lee, et al., 2013) further inspires the idea of collecting data using the phone’s hardware, which is, of course, a much more comfortable option for the user. It also results in the user being much less conscious of the analysis, since the phone simply runs in the background and causes less disruption with the user’s activities whilst collecting and analysing data.

As such, (Lee, et al., 2013) inspires the idea of utilising the phone’s hardware. We can note that the physiological reactions sensed in (Ertin, et al., 2011) can now be sensed using current mobile phone technology. There are methods to calculate one’s heart rate, such as in the application “Stress Check” (Azumio, 2012). This application uses the phone’s camera and flashlight to detect one’s heart rate. We can detect respiratory function using the phone’s inbuilt accelerometer to detect a rate of displacement on one’s lung area, hence allowing us to accurately calculate changes in volume of air within one’s lungs (Chandrasekeran, 2010). (Ertin, et al., 2011) also detects skin temperature, which can be found using the phone’s thermometer. All these inbuilt hardware render the external hardware used by (Ertin, et al., 2011) as redundant. The advantage that the external hardware offers, however, is that people still claim accuracy, however (Ertin, et al., 2011), despite using dated technology, still determines one of the highest accuracies out of all the solutions. This can be attributed to many factors, with the inaccurate nature of using the internal hardware of a phone being one of them.

Whilst both (Ertin, et al., 2011) and (Lee, et al., 2013) uses static information for its inferences, (Ertin, et al., 2011) has a 90% accuracy rate for 20+ participants, and (Lee, et al., 2013) has an overall accuracy of 60%, with variations of ±5% due to different phones. A phone’s internal hardware can cause inaccuracies specifically due to its physical placement upon collecting of data, as well as the design of the hardware itself. (Ertin, et al., 2011) conveniently places the hardware on the person’s chest. Whilst this is possible to achieve using internal hardware, it is highly inconvenient for one to place their phone on their chest – this would take the element of convenience by using a mobile phone’s internal hardware to create less interference with their daily activity. (Lee, et al., 2013) only uses the common camera and microphone hardware in a phone. Depending on where it is positioned, it can be used conveniently for analysis. However, at the same time, there is a degree of inaccuracy, depending on at what distance the phone is placed at during its inaccuracies. Whilst the hardware is of relatively good quality, it is found that dedicated hardware to these purposes are still of better quality.

One significant advantage that each of the solutions offer is the use multiple parameters to determine its results. These use algorithms that can weigh the importance of each of the parameters and make a decision from there. This is highly advantageous, as it allows us to hone in on the specific reasoning behind the physiological reactions. We must remember that physiological reactions can apply to more than one type of response (Carlson, 2007). The use of multiple parameters offers a distinguished combination of symptoms which can be more accurately associated with stress. However one disadvantage of these applications is that these are static algorithms that do not account for differences in human nature from user to user. These algorithms can be improved so as to cater for users more accurately.

We must draw a further parallel between SocioPhone (Lee, et al., 2013) and AMMON (Chang, et al., 2011) as two of the key applications used to detect changes in speech during times of emotional stress or distress. As mentioned before, SocioPhone has produced an accuracy of 60±5%. On the other hand, AMMON produced an accuracy of 93.6% accuracy. Whilst these both are dependent on speech variations

AMMON in particular has used a balanced data set to ensure a 50% degree of accuracy. Balanced data is essential as training data. Whilst we can note that data that is biased towards one particular result can still produce results of decent accuracy, it can fundamentally be proven that balanced data will produced more accurate results (Wei & Dunbrack Jr, 2013). Thus, we recognise this as one particularly important characteristic regarding machine learning.

(LiKamWa, et al., 2013) has a contrasting aspect to its ability to cater for multiple users. It uses aspects of machine learning over a period of time to increase the accuracy of the application. Using only static inferences, the application offers a very low accuracy (66%), whilst post-training, the application was able to provide an accuracy of 93% to the two users (LiKamWa, et al., 2013). This is a highly significant change attributed solely to concepts of machine learning, and hence a large factor to consider in achieving our own personal goal. The application actually uses less reliable methods of detecting moods than as per previously mentioned, but achieves the same degree of accuracy as those applications.

The application uses a supervised learning algorithm referred to as Sequential Forward Selection (SFS). SFS involves machine learning via regression. Unknown data inputs estimate and produce an inference using regression techniques. The algorithm is used hand-in-hand with a personalised data model to create an even more accurate data model on a case-by-case basis. Whilst the personalised data model can be used by itself to create inferences, it also involves a much longer time-frame for the model to reach its potential. This issue is solved by complementing with the SFS algorithm, which acts as a “one-size-fits-all” model to give the application a starting idea of how to process the information, as opposed to developing it from scratch (LiKamWa, et al., 2013). The following graph gives us the best idea of how the application evolves over the 60-day learning period.

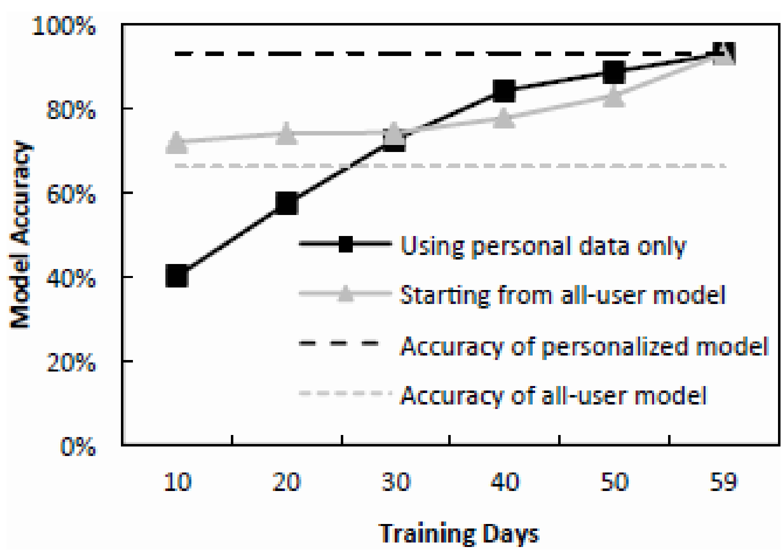

Figure 1 – Improvement in accuracy over a 60-day period due to machine learning (LiKamWa, et al., 2013)

The initial concept of how to detect mood, however, is not as superior for the purposes of detecting stress. The application is not reliant on physiological stress responses, but rather stress responses based on direct phone activity. This leads to a low degree of accuracy; as such activity widely differs from person to person, with very little overlap of common activities for all potential users. Accuracy is further lowered since people’s activities vary upon the onset of stress (Carlson, 2007; Segal, et al., 2014; American Psychology Association, 2014).

As such, the main aspect of the application to take as inspiration is the concept of supervised machine learning to create an accurate model. Whilst the methods of collecting data are respectable, other applications have more accurate results from methods of collecting physiological data to detect the onset of stress.

(Logan, et al., 2007) is another 7-year old application which has still proven to be very resourceful. The big picture of our aim involves the dangers that stressors and stress responses bring to an individual, and the technology used in (Logan, et al., 2007) can be used to address this issue directly. We can also use these methods to determine levels of stress. This methodology still uses external hardware to obtain the raw data regarding blood pressure. However, the application is used for the processing of this data, after it is received from the external hardware. The information is then sent to a central server for notification using mobile technology.

The application uses one of the most direct symptoms of stress responses possible, however is currently not very feasible to achieve only using a phone's hardware. Whilst there have been developments, such as the enterprise iHealth Wireless Blood Pressure Wrist Monitor (iHealth, 2012), none have progressed past the stage being restricted to using external hardware to determine this parameter accurately. As such, it is currently infeasible to achieve our goal until the necessary hardware is incorporated into mobile phones.

The application, however, inspires the idea of further refinement of the application, by sending the data to relevant places. There is not much data available that has been collated and sourced from a mobile phone for our purposes. This application, however, takes this data. Hence, this solution proves useful for the collection of data, as well as the benefits of alerting the appropriate people, for monitoring purposes, in a non-invasive manner.

Whilst not all projects studied used machine learning techniques, it is still important to recognise the importance of it in the research and projects thus far. Another namely example we can look at shows how the different methods of machine learning act to give varying accuracies of results (Lu, et al., 2012). This was especially important in ensuring the accuracy of the product in determining stress from a multitude of different symptoms. They implemented three models, using various techniques including the use of the linear SVM. Each model used machine learning, but varied between levels of supervised and unsupervised learning.

Machine learning is particularly helpful in determining use-cases that are tailored to the user – especially if unsupervised or semi-supervised learning is allowed. In this case, semi-supervised learning is used to develop their model. Their "supervised adaptation" involves "a user explicitly contributes labelled data for adaptation" whereas the "unsupervised adaptation", i.e. self-training, involves "leverages self-train technique by utilizing unlabeled data" (Lu, et al., 2012). What we see when we compare two of their models, where one was universal (fully supervised) and the other was personalised (fully unsupervised), was that the personalised one was the most accurate model, whereas the universal one was the least accurate. As an example, for an outdoor environment, the models achieved accuracies of 77.9% and 66.6% respectively (Lu, et al., 2012). We also note that the variance of the results for the supervised learning is much higher, as seen in the diagram below.

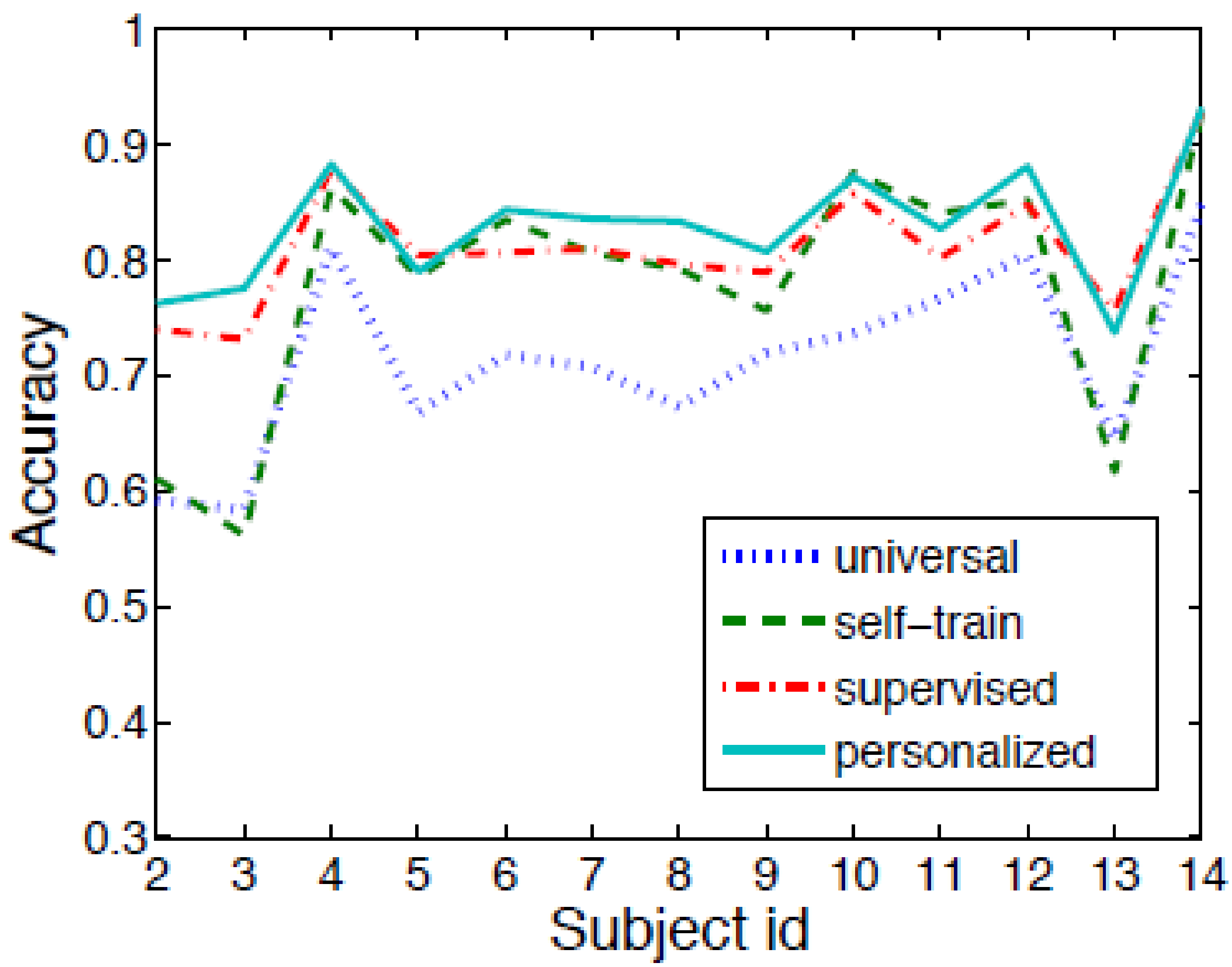

Figure 2 – Varying accuracies dependent on level of supervision in StressSense (Lu, et al., 2012)

Thus, we cannot reject the notion of unsupervised learning completely. Whilst it is very risky to use (Flach, 2012), it also produces the most correct results (Lu, et al., 2012). The risks involved are if:

- Algorithms are implemented correctly, and
- if the users use the application correctly, and do not feed it bogus data – be it intentional or unintentional

2.1.2 Non-mobile-application solutions, and data collection methods

There are many applications that do not include using a mobile phone that are used to detect stress – most of which are as effective, if not more than current state-of-the-art technologies using a mobile phone. Good examples to illustrate this are as per below.

Mood Meter [7] was not a mobile-phone based application, but still uses technology that mobile phones are currently able to use today. A camera is used to detect people's "smiles", and rate them on a scale according to the "Shore framework" [9] – a geometric algorithm that detects a smile, and its intensity, with a high degree of accuracy even from a distance [7]. The data is then processed and sent to a central server for collating and further processing.

The use of a heart rate monitor to detect stress is a particularly useful one, with one of the main symptoms of stress being an increase in heart rate. We must note that the type of stress detected is of all kinds, and even beyond the scope of research. However, we must appreciate the advantages offered before us. One particular application involves sensors of minimal obstruction (Choi & Gutierrez-Osuna, 2009). The sensors are used to collect data regarding a variable heart rate over time. This is statistically analysed to determine changes in stress levels.

Similarly, blood pressure can also be analysed using a blood pressure cuff. This, complemented with changes in heart rate, gives us ample information, as seen before. Experiments involving these signals, as well as changes in eye activity have been done in (Sharma & Gedeon, 2013). Through the watching of films, stresses were incited. From here, the changes from the person's different states were matched up with the bodily activity of the person. The data underwent statistical analysis, and machine learning algorithms were applied to determine stress. These, once again, produced meaningful results.

Cardio-respiratory systems are most evident in continual physical stress in particular, such as physical shocks from running, or simply having a bucket of ice poured over your body. However, experiments have been done to observe symptoms due to mental stress (Choi & Gutierrez-Osuna, 2010). As a result, experiments have been done so as to analyse oscillations not only in heart rate variability, but also in breathing patterns and sweating. This is performed in a similar manner to that as of above, and has produced significant results that are actually positive.

Ideally, the developed solution would involve remote data collection. Frameworks have been produced to detect heart rates remotely i.e. without physical contact. This is done through a number of physiological reactions associated with the heart rate, such as reading the human face (an extremity of the body) using thermal infrared imaging (Bousefsaf, et al., 2013). This produced significant results that actually determined changes between stress and a lack thereof.

We must note that the extremity of the human body is a particularly good source of detecting changes in heart rate. This is due to the nature at which blood collects and empties at this particular point. Whilst the human body extremities also include smaller features such as toes and fingers, the face is an optimal place to analyse blood flushes. Most blood rushes to this extremity due to the size of the head in comparison to other extremities. Thus, this produces the largest changes in visual flush as well as temperature due to blood flushes. We must also remember that the rate of flushes in and out of the extremity is directly proportional to one's pulse i.e. their heart rate. Thus, thermal imaging, which can detect small changes in colour due to blood flushing (Bousefsaf, et al., 2013), namely, can be used effectively to detect the nature of one's heart rate.

We have seen good methodologies to look into regarding what to look out for with stress. Throughout all papers, we notice one of the ways of detecting a stress symptom is via detecting heart rate volatility (HRV) (Choi & Gutierrez-Osuna, 2009; Sharma & Gedeon, 2013; Choi & Gutierrez-Osuna, 2010; Bousefsaf, et al., 2013), with each research paper deeming different importance on it. As aforementioned, this is extremely important due to the strong correlation between one's heart rate and their emotional state i.e. their stress levels. In particular, those that used HRV to detect stress levels ended up with a very positive result that showed strong significance.

In one particular instance, machine learning is still used. We have mentioned the advantages of machine learning previously. In the case of (Sharma & Gedeon, 2013), universal optimised search algorithms were used in their SVM and artificial neural networks (ANNs). All test subjects used were between the age of 18 and 30. What we noticed is the machine learning algorithm used was not significant enough of a factor to change the accuracy. The more significant case involved method of detection. Methods involving detections in physical reactions labelled as "primary stress signals" returned an accuracy of 95%. However, when using electroencephalography (EEG) to detect physiological changes, the accuracy was determined to be 91%. With all factors deemed constant, we see that physical symptoms are, indeed, very powerful in determining stress.

We also note that in this experiment, the heart rate was not a key determining factor as a data source. This was a very intricate set up with a large amount of excessive equipment, since it was to simulate a virtual environment. A lot of data samples were taken from various sources, allowing the increased accuracy without the high dependence on HRV – something that is not common.

In the other three cases, we use a less complex form of machine learning and less equipment than the previous example. We notice that despite the complexity of algorithms, we end up with a lower accuracy. One example is 83% accuracy when still detecting heart condition, muscle tension and respiration (Choi & Gutierrez-Osuna, 2009). This used a non-linear Volterra series to establish stress levels[3]. In the same way that the charge of a battery deteriorates over time, we note that the heart signals observed over time from the heart monitor has varying importance. This procedure of detecting stress, in particular, places a higher importance on HRV than the other factors. Therefore, we can appreciate machine learning as an important factor, as well as the other factors that determine stress that are not weighted as importantly.

Respiration was the second most prominent factor in these determinations. Namely, in non-mobile-phone applications, we notice its use in (Sharma & Gedeon, 2013) as a very good example. However, this uses an undiscussed hardware – ECG and galvanic skin responses (GSRs). These are highly accurate pieces of equipment that can be used to detect one's conductivity of the skin. This has a strong correlation with sweating, a symptom of stress. This is due to the high electrolyte content of sweat – hence dry skin has a much higher resistance than wet skin. The difference is significant enough to detect a change in.

Again, we've noted the accuracy it has produced, and can attribute a part of that to the many factors that it takes into account. We must still consider the importance of the use of the SVM to create further accuracy, as well as the detection via eye-ball – something that no other relevant paper has considered and produced a strong accuracy.

[3] A Volterra series is an algorithm used to predict the decay of a device's ability to hold electrical charge (Volterra, 1959)

Overall, the accuracies determined throughout these experiments have a much higher accuracy than that of those described in section 2.1.1 – up to 95% accurate, versus a maximum of 83% respectively. Mobile application solutions have restrictions due to the mobile phone system, such as limited processing power, that the non-mobile phone solutions don't have. In particular, we must note that specialised external hardware provides a significant advantage in accuracy over those of a mobile phone. Their usage is also less applicable to the practical world, as the equipment is optimised for simulations rather than day-to-day use. This is a significant disadvantage to a lot of non-mobile solutions and, thus, is a further inspiration for our proposed solution.

2.1.3 Research regarding stress factors generally

There are a number of articles that describe the changes in emotions and stress due to various factors. A few different cases were analysed for the sake of diversity and a better understanding of the problem and potential solutions at hand. These were also assessed for relevancy in applying to our proposed methodology.

One particular study involves the changes of the human emotional state based on social media patterns, and resulted in the development of "Moon Phrases" (De Choudhury, et al., 2013). This is increasingly relevant in this current time, considering the amount that social media is used. One of the key factors that can be used to detect emotional disturbances is changes in linguistics. In particular, people express their emotions through Facebook, Twitter, etc. when interacting with friends or strangers. Initially, 6 subjects had their social media use analysed with feedback as to their current mood at the time. From here, a framework was developed which isolated linguistic factors to determine the mood in which a person was in based on their activity. Whilst it has not achieved its aim in acting as an intervention mechanism, it has still progressed towards it through accurate identification of emotional wellbeing and has the potential to have a positive impact on reducing negative behaviour of heavy social media users.

Whilst our main focus has been on physiological and physical reactions thus far, there are also deeper biological reactions to stress as well that can be analysed. It has been reported that prominent stressors that cause this include general stress, heat shock, cold shock, cytoplasmic stress, and starvation. The experiment involved the extraction of the biosensor (a trigger for our biological stress responses) and seeing under what conditions they survive. A study of this was conducted successfully (Abraham, et al., 2011)

In terms of capabilities, phones are not equipped with all the hardware required for the analysis which we require currently. One example is obtaining biological data to help determine stress levels (Abraham, et al., 2011). For this reason, one disadvantage of the use of a mobile phone is that it is more probable that we achieve a better accuracy in all cases with other hardware.

2.2 Conclusions and Refined Aim

Detecting the onset of stress using a mobile phone involves capitalising on stress responses from the user, measuring their intensity and making a judgement. As mentioned before, we are in need of a method of stress detection that acts dynamically, and responds to stress immediately as opposed to only upon user request.

A key differentiator between people is what we respond to in a stressful manner, and what the physiological and mental reactions are. There are many responses that are relevant to people, and we can only say whether it is or not on a case-by-case basis. However, there are stress responses people experience are involuntary, and common amongst the majority of people. Hence, stress responses, which are a unique combination of physiological reactions, can be detected and identified strictly as stress. We are, of course, familiar with common stress responses. Whilst there is a large variety, examples of some common symptoms (Carlson, 2007; National Institutes of Health, 2014; Segal, et al., 2014) include:

- Changes in vocal pitch and amplitude
- Erratic movement, e.g. pacing up and down a certain distance
- Increased heart rate
- Perspiration via palms
- Increased blood pressure
- Brain damage – specifically learning and memory
- Weakened immune system, and associated nervous system

Our reviews have exposed information regarding physiological, physical and biological symptoms of stress, as well as how they are detected and strong determinants of whether one is exhibiting stress symptoms or not. We also note that some methods have produced stronger accuracies than others. Whilst accuracy is one of the most important factors, we must also consider other factors such as practicality in the real world. Such practicality for the user includes portability and ease of use.

We have decided to use a mobile phone application to take advantage of the portable hardware within the phone itself. However, the disadvantage of this is the inability to use the intricate software offered by some of the non-mobile solutions. Thus, we must investigate what hardware offers a high accuracy and are integrated into a phone.

Are You Stressed?

For a start, we noted that biological symptoms (Abraham, et al., 2011) cannot be detected with a phone, as discussed above. Thus, the main concerns for our conclusions will revolve around physiological and physical symptoms of stress. Although these offer less accuracy at times, under certain conditions, they are the most practical symptoms to make use of.

We notice that heart rate is an extremely good indicator of stress, both mentally and physically. This has been mentioned previously in the literature review, let alone within the papers themselves on numerous occasions. A phone has the potential to recognise one's heart rate using the camera[4]. Such an example is its application in Azumio's "Heart Rate Monitor". Whilst this development is somewhat of an innovation based on an undedicated hardware (the camera), the logic to deduce one's heart rate through imagery is correct. Thus, we will be able to take use of this signal using a mobile phone.

Another key signal we would like to observe is respiration symptoms due to stress. Respiration occurs in the form of cardio-vascular activity or perspiration. Again, papers that used respiration techniques had a furthered accuracy which we would like to take advantage of.

Perspiration can be measured in a phone using the hygrometer, which we will discuss later in this thesis. Again, this is not the dedicated purpose of the hygrometer[5], however the logic behind its use is correct and, provided we can provide a proof of concept to ourselves for its application, and it is a feasible option to use for detecting perspiration.

Lastly, we take note of two mobile applications that make use of speech patterns and their associations with stress. By itself, speech has the potential to provide one of the highest accuracies. We have also noted that libraries have been developed regarding speech patterns – in particular, specifically for stress (Chang, et al., 2011). This is expected behaviour, with phones having a microphone dedicated to detecting speech for one of their primary uses, phone calls, and other uses such as recording. Thus, this is the last physical stress reaction that we will take into account as inspiration, based on the literature reviews.

However, we note that there are many other stress symptoms that have not been taken into account at all in our research. Due to the new nature of this topic, we cannot expect all symptoms have been

[4] Written as of June 20th, 2014. Future developments have occurred since this conclusion, to be discussed later.
[5] A 'hygrometer' is defined as "a device for determining the humidity of the atmosphere" (HowStuffWorks, 2009). It can also be used to measure the moisture of one's palms when holding the phone.

taken into account thus far. Such namely stress symptoms are some of those mentioned above. One in particular that will be discussed later is one's erratic movement, which can be detected using a phone's accelerometer. Despite a lack of investigation in previous literature reviews, other sources indicate subconscious movement is a body's way of minimising mental stress (Carlson, 2007; CalmClinic, 2014). This theoretical knowledge has resulted in a decision to investigate this symptom, which is viewed as a knowledge gap.

Current methodologies that are successful account for specific symptoms. The aim of the next part of the chapter is to explain the current solutions to the problem at hand, what they offer for the detecting the onset of stress, and what each of the latest resolutions are lacking from the perfect solution.

3. OWN WORK

After the research conducted via our literature review, we are then ready to set a proposal forward of potentially good ideas that will help achieve our aim. We will also discuss the manners and methods in which we go about achieving our goal, as well as changes in our proposal that have led to a differing solution upon realisations and circumstances.

3.1 Proposal

3.1.1 Deductions

The aim of this chapter is to show a plan of how to fill the voids of the problem through other means. We have discussed where the current applications and solutions stand, and so from here we can utilise the benefits to fulfil the ultimate objective. We can combine these benefits, with slight additions, to form a long-term project. The current goal is to lay down a strong framework that can be built upon for future generations of research. The overall goal is to minimise stress responses and the dangers associated with stressors and stress responses.

From our previous analysis, we can deduce requirements of the following:

- We are in need of a system that detects the onset of stress in a convenient manner
- A mobile phone has been deemed the most appropriate equipment. It is portable, and allows easier detection of the onset of stress with less conscious effort. Mobile phones are also equipped with a lot of helpful hardware, and are equipped with more as time passes due to technological advancement. Due to the capabilities of a mobile phone, we will eliminate the need for external hardware, as most are used to detect stress responses are now integrated into the latest phones.
- Machine learning algorithms are a key part of the project, in order to ensure accuracy for all users – there is no one size fits all, and there is always room for improvement due to the varying nature of stress responses from different users.
- Mobile phones also have abilities to communicate, not only with the user, but globally by utilising data connections such as Wi-Fi and mobile data networks. Hence, we will require a form of persistence to collect data that can easily be sent by users of the application. This data can assist with the machine learning process and, combined with the initial data provided,

allows a unique form of semi-supervised learning that has not been used in any state-of-the-art applications to date

A person experiencing stress responses is not necessarily acting rationally. We require a solution that will collect our data through natural means. This includes when the phone is on standby next to the user or in their pocket, when we are using our phone for remote video or vocal communication, or simply through interactions when using other applications such as games, the Internet, video viewing, etc.

A mobile phone is a convenient gadget to use, with over two thirds of Australians owning a smart phone (Stafford, 2013). In particular, we will initially target specific stress responses and expand on these further. Stress responses to analyse include heart rate, vocal communication, and use of sweaty palms and a person's rate of pacing. Each of these has been determined as a stress response by (Carlson, 2007; WebMD, 2013; National Institutes of Health, 2014). It is a unique combination of physiological activity that is most often related to mental stress. These are also feasible symptoms to measure using the latest mobile phone technology. The associated hardware with each of these responses is the camera and flash, microphone, hygrometer and accelerometer respectively. With technological advancement and time, the project will be able to expand further and detect other stress responses, such as detecting blood pressure using only the phone's internal hardware.

Concepts of machine learning are sure to be adopted to increase accuracy. Initially, the application prototype will consider binary classification of experiencing an onset of stress or not. Future developments will involve a non-binary classification of how much stress is experienced by the user, on a scale involving more than two options. In considering binary classification, we will start with a simple Bayesian model, and change the model according to what results we start to obtain, as well as what algorithm will be used. The model could be SFS, as mentioned in (LiKamWa, et al., 2013), or random forest decision tree, as discussed in (Flach, 2012). The application will also be fed test data, to provide a learning environment which the application can learn around using the appropriate algorithms. This will ensure a more acceptable degree of accuracy, whilst improving to perfection over time. As we will discuss next, we will actually adopt a linear Support Vector Machine (SVM) to ensure efficient learning processes with maximal accuracy.

We are also well aware of the communication abilities that a phone has. Being able to send data globally allows global users to send their data regarding physiological stress signals, as well as whether the determined result was correct or not. We can thus use the concept of a persistence model to collate useful data sourced solely from a mobile phone, and use this in future research. Furthermore, whilst the initial process of supervised machine learning involves having a set of test-data, we can also collect more initial data for future users by collecting from the current application users themselves. In having more data, we are adopting a semi-supervised learning process (Flach, 2012) - we are using data that is sent without interference from the developer. Whilst we run the risk of false data being sent for use, the risk of corrupted test data is minimised using the original test data, and giving it more weighting when used in modelling.

Thus, upon research, we have determined a seemingly perfect solution that has not been exercised yet. A proof of concept has been partially developed, where we are testing the hardware and the degrees of accuracy to which the hardware hold true.

With the progression that stands as is, as well as a partial proof of concept that has been developed, we now have the following to consider over our time period of Tuesday May 27th 2014 – Friday 3rd October 2014, which is described in Figures 3 and 4.

	Name	Start	Finish
1	**Proof of concept**	**28/05/14 8:00 AM**	**6/08/14 5:00 PM**
2	**Use of hardware**	**28/05/14 8:00 AM**	**15/07/14 5:00 PM**
3	**Ensure accuracy of accelerometer**	**28/05/14 8:00 AM**	**6/06/14 5:00 PM**
4	Assess how to stabilise accelerometer throughout movement	28/05/14 8:00 AM	2/06/14 5:00 PM
5	Implement algorithm to stabilise values	3/06/14 8:00 AM	6/06/14 5:00 PM
6	Ensure accuracy of hygrometer	6/06/14 8:00 AM	10/06/14 5:00 PM
7	**Use of microphone**	**11/06/14 8:00 AM**	**26/06/14 5:00 PM**
8	Create algorithm to find pitch of voice	11/06/14 8:00 AM	18/06/14 5:00 PM
9	Implement algorithm for voice	19/06/14 8:00 AM	26/06/14 5:00 PM
10	**Use of camera**	**26/06/14 8:00 AM**	**15/07/14 5:00 PM**
11	Research detecting changes in screen	26/06/14 8:00 AM	1/07/14 5:00 PM
12	Implement a change detection API for application	2/07/14 8:00 AM	7/07/14 5:00 PM
13	Regulate results to detect heart rate	8/07/14 8:00 AM	15/07/14 5:00 PM
14	**Statistical collection of data regarding stress responses**	**16/07/14 8:00 AM**	**6/08/14 5:00 PM**
15	Analyse stress via voice	16/07/14 8:00 AM	18/07/14 5:00 PM
16	Analyse stress via heart rate	21/07/14 8:00 AM	23/07/14 5:00 PM
17	Analyse stress via sweaty palms	24/07/14 8:00 AM	28/07/14 5:00 PM
18	Analyse stress using pacing	29/07/14 8:00 AM	1/08/14 5:00 PM
19	Identifying strong changes between stress responses	4/08/14 8:00 AM	6/08/14 5:00 PM
20	**Creating stress detection algorithm**	**7/08/14 8:00 AM**	**22/08/14 5:00 PM**
21	Summarising strong changes from statistical analysis	7/08/14 8:00 AM	8/08/14 5:00 PM
22	Use summarised data to create algorithm	11/08/14 8:00 AM	18/08/14 5:00 PM
23	Implement basic algorithm	19/08/14 8:00 AM	22/08/14 5:00 PM
24	**Create application**	**22/08/14 12:00 PM**	**28/08/14 5:00 PM**
25	Create GUI	22/08/14 12:00 PM	25/08/14 5:00 PM
26	Integrate application to respond during phone call	26/08/14 8:00 AM	28/08/14 5:00 PM
27	**Testing**	**29/08/14 8:00 AM**	**8/09/14 5:00 PM**
28	Collate data from multiple users	29/08/14 8:00 AM	5/09/14 5:00 PM
29	Process data in same fashion	3/09/14 8:00 AM	8/09/14 5:00 PM
30	**Refine algorithm**	**9/09/14 8:00 AM**	**16/09/14 5:00 PM**
31	Summarise statistical data	9/09/14 8:00 AM	10/09/14 5:00 PM
32	Implement binary machine learning algorithm	9/09/14 8:00 AM	16/09/14 5:00 PM
33	**Finalise application**	**17/09/14 8:00 AM**	**24/09/14 5:00 PM**
34	Package application and data	17/09/14 8:00 AM	17/09/14 5:00 PM
35	Test on phone	18/09/14 8:00 AM	24/09/14 5:00 PM
36	**Evaluation**	**25/09/14 8:00 AM**	**3/10/14 5:00 PM**
37	Summarise new data	25/09/14 8:00 AM	25/09/14 5:00 PM
38	Comparison between last 2 test cases	25/09/14 8:00 AM	30/09/14 5:00 PM
39	Conclusion	1/10/14 8:00 AM	3/10/14 5:00 PM

Figure 3 – Planning until preliminary demonstration

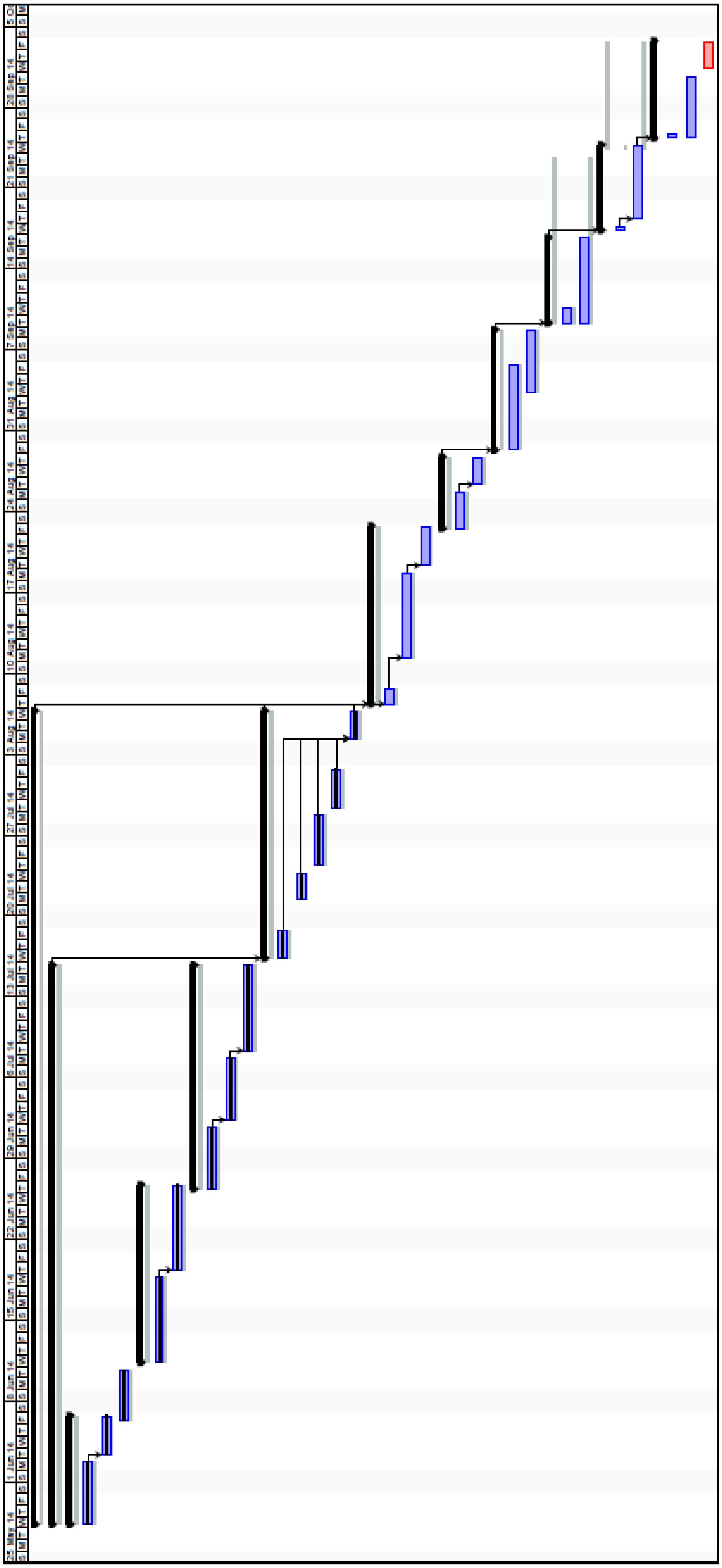

Figure 4 – GANTT chart representation of plan

3.1.2 Breakdown

The aim of this section is to justify the necessity for the breakdown of our project.

A proof of concept is especially important in research, as this will give a true assessment of what we can do with mobile phone technology for the purpose of the project. Categorically, we can break the assessment of each of the hardware into determining the accuracy, and how functional it is for the purpose. Whilst we are aware of how the hardware can be used theoretically. However, the investigation in the allocated time period will give a true assessment of how it can act functionally.

The hardware of a phone can be used for many purposes. We must consider how to use the each hardware's API works towards the final goal. In some cases, we have left minimal time, as the algorithms required to collect relevant data are not complex. However in other cases, namely the camera as a prime example, require more complex algorithms, since we must detect dynamic changes in a hardware which we do not normally consider. The camera’s functional purpose is to capture images for storage. However, we are using the camera to simply analyse each capture and take relevant data dynamically. For such purposes, we have allocated more time.

We then finalise the functionality assessment using a statistical analysis of collected data when using the hardware for the specific purpose. This data is not only used for a furthered proof of concept, but as an initial set of test data to help build our initial starting algorithm, as per our semi-supervised learning process.

From here, we can create our stress-detection algorithm using certain regression techniques, and potentially implementing machine learning concepts – in particular, the Bayesian model for binary classification. Considering we are using a small sample of data at this point for testing purposes, we do not expect the model to be completely accurate. We also do not intend on implementing unsupervised learning at this point.

The testing process is a supervised method of collecting further data to assess how effective the application works. This will be used in our evaluation process when assessing the effectiveness of the application without implementing unsupervised machine learning processes. This data will also be used as raw data to help assist with the supervised learning process.

The new data is summarised, and from here the algorithms can further be refined. Hence, an efficient machine learning process is integrated into the system. Once this is complete, the application is ready to be used. The application is packaged, with the collated data incorporated for supervised learning purposes.

We then re-collect the data through having a final testing session with the same sources, as well as other people. This will end our collection of data for evaluation, and the information will be summarised.

3.1.3 Changes to proposal

We also notice that there is new hardware to support the human heart rate (Samsung, 2014), namely in the Samsung Galaxy S5. This dedicated hardware is much more effective than the camera in detecting the heart rate, leaving our idea of using the camera and respective flash light as a primitive method of detecting heart rate. However, we are imposed with two restrictions:

- The device hardware is currently specific to Samsung, and requires the Samsung S Health SDK.
- All devices that are equipped with the heart rate monitor do not come equipped with the necessary SHTC1 chip.

The heart rate monitor is a very recent development, with the release of the first one in the Samsung Galaxy S5 occurring during our project development (GSMArena, 2014). Thus, our proposal is flawed in using an inferior technology such as the camera. Whilst there is potential to monitor heart rate on a phone (especially being one of the most powerful tools), we require it to be available on more phones (that are affordable). Due to hardware restraints, we choose to ignore these features for now, and these will be directed to future work.

3.2 Specific Learning

3.2.1 Android API

Further research was conducted regarding elements of the Android API. To proceed, we investigate specifically how to utilise each hardware device. We use Java 8 SE, with the Android SDK providing further libraries for our use.

The accelerometer and hygrometer both utilise the SensorEventListener interface that comes standard in the Android libraries, as will be discussed in the implementation. This allows us to implement what happens once the respective sensor detects an event.

An accelerometer delivers three useful types of information, as presented below.

Sensor	Sensor event data	Description	Units of measure
TYPE_ACCELEROMETER	SensorEvent.values[0]	Acceleration force along the x axis (including gravity).	m/s^2
	SensorEvent.values[1]	Acceleration force along the y axis (including gravity).	
	SensorEvent.values[2]	Acceleration force along the z axis (including gravity).	

Table 1 Data collected from accelerometer (Google, 2014)

A hygrometer delivers information, as presented below. The percentage is a direct measure of the detected, relative humidity i.e. "the amount of water vapor in the air relative to what the air can hold." (Education Portal, 2003-2014).

Sensor	Sensor event data	Units of measure	Data description
TYPE_RELATIVE_HUMIDITY	event.values[0]	%	Ambient relative humidity.

Table 2 Data collected from hygrometer (Google, 2014)

We note that the sensor delivers values in our 2 other values in the presented array. These are ignored, and are of no use to us. However, for future progression, we will keep a log of them in the event they prove useful.

Furthermore, we use the AudioRecord object to record any speech made into the phone. Whilst this is reading, the data collected into a buffer is a representation of the amplitude of the detected sound. This occurs until it is directed to stop, and from there we can process the data in a similar manner to the other collected data.

We also take note that the Android API is used to activate an activity. This is mainly done to create displays. In particular, we create methods of notifying and providing actions points to the user. These include the use of:

- Toasts
- TextViews
- Buttons
- Dialogs

We then use data stored aside, combined with new data to assess whether the person is currently stressed or not. These are done using implemented machine learning algorithms using standard Java classes, which will return data to the main activity and show displays accordingly, as well as add to our saved data.

3.1.2 Machine learning algorithms

Upon research, we have decided to use a form of binary classification. There is a clear difference between when someone is stressed and not. Our aim is not to assess to what degree of stress they are experiencing, but more so whether the levels of stress they are experiencing are irregularly high or not – clearly there are only two options to consider in this case. Most other papers simply consider binary classification.

Similarly, other papers consider the use of Support Vector Machines (SVM). This is a commonly used tool for supervised learning using binary classification. Thus, for our method we will have to collect data simply using the sensors. This data will be used as our supplied data for our supervised learning algorithm, in which the device can accurately deduce whether one is stressed or not based on historical data.

However, we do not intend on using complete supervised learning. This is not a good idea due to the varying nature of different stress responses from person-to-person. Thus, we must incorporate levels of unsupervised learning. This will allow data to be added upon each use of the application to detect stress. The data and the classification will be recorded, and added as data for the SVM. Thus, we are implementing a semi-supervised learning process for the application.

3.3 Process

3.3.1 Data Gathering

An initial application was tested to utilise the previously discussed learnings and gather data of all forms. Whilst this was not an elegant application, it was a simple creation to gather data. Whilst the implementation of this is independent of device, it is very important to ensure the hardware itself exists in the device.

The device we have chosen is the Samsung Galaxy S4. We must note in particular that the Samsung Galaxy S4 is one of the few devices that are equipped with the SHTC1 Sensor Chip, developed by Sensirion. That is, there is a temperature and humidity sensor in-built into the phone. Thus, we are equipped with the necessary hardware for the three symptoms we intend on detecting i.e. through a:

- Hygrometer (from the SHTC1 chip)
- Accelerometer
- Microphone

The application used two SensorEventManagers at the time, each with their own SensorEventListener that tapped into the respective hardware. This produced the expected results, using our chosen device. The data was also logged, whereby it could be extracted and inferences could be made.

To formalise our processes of data collection, the following method was used:

1. An apparatus of a phone connected to a portable laptop via USB cable is set up
2. A bash process was set up to record all log files:

   `$ adb –d logcat > output.txt` [6]

   We note that we can pipe our initial command to output based on tag using the 'grep' command, before forwarding the information to another file
3. 2 people are taken, and kept in a calm and relaxed environment. These people are to use the application in the following steps.
4. The first user is to hold the phone at a one metre distance from their mouth before speaking into it for one minute, with an activated microphone
5. Step 3 is repeated, with the user now elevating their voice

[6] Done on a bash terminal, Linux Ubuntu 14.04 LTS

6. The user is then asked to walk on a straight plane for one minute
7. The user then is asked to pace back and forth over a 7 metre space for one minute, with an activated accelerometer
8. The user is then asked to simply hold the bottom half of the phone, around the sides, for another one minute, with an activated hygrometer
9. Step 7 is repeated, after dampening the user's hand lightly
10. This process is repeated 3 times, followed by a repetition of all steps for the other user.

3.3.2 Manual Inferences

We then take away the information and, through statistical analysis, we assess whether these instruments are able to perform the necessary operations. Furthermore, we check for whether it is possible to distinguish between the simulated stressed and unstressed environments.

Simple averages are used on the logged data to determine what kind of values we are dealing with. We find this suitable due to the relatively low variance with few outliers.

Data	'Stressed' state	'Not stressed' state
Amplitude of voice	51.9	168.4
$\frac{\vert \Delta a_z \vert}{\vert \Delta a_y \vert}$	0	8
$\frac{\vert \Delta a_z \vert}{\vert \Delta a_x \vert}$	0	8
Relative Humidity	65.1	68.2

Table 3 – Assessment of changes in numbers in stressed vs. unstressed environments[7]

3.3.3 Algorithm Construction

As discussed before, we are using a linear SVM. A series of test data, namely gathered from the simulation, is used to create a learning environment that the application can create its initial SVM.

We note that we have also implemented a supervised learning aspect by allowing the user to input whether he or she is stressed or not. Therefore, as the application is used, the device will learn more

[7] '$\vert \Delta a_n \vert$' refers to count of times a significant change in acceleration was detected, in the x, y or z direction respectively

about the person. Therefore, data points will be added to the SVM that are tailored to the user, and the algorithm will change according to the skew.

We notice that the data produced has outlier values on occasion. The reasoning behind this cannot be identified, since it cannot be replicated with all other factors held constant – if anything, one could say the odd values occur at random times. Nonetheless, we look to eliminate these values.

In accordance with our values from Section 3.3.2, we see that there are definitive values for which we can associate with being stressed based on each of our three sensors. Yet the incorrect data can skew our data incorrectly. The range of values found for each hardware input can be used to determine whether the values obtained are an outlier or not.

Thus, we refine the machine learning process to accommodate for the removal of the outliers. Whilst the data remains, the current algorithm will not consider these values when deducing new values.

3.4 Implementation

From here, we are ready to implement our solution – the 'StressTester'. As per the nature of a native Android application, the solution was developed in Java, with the Android SDK. Outlined below are the details of the implementation. This is categorised by the package names used. A summary UML diagram to show the structure between the classes is also included. Screenshots of the implemented views have been included in the appendix.

3.4.1 com.hari.se4911.stresstester

This package contained two sub-packages for the recorders and the results, as well as a class that the application worked around. The application requires an Activity to work with, so as to do its display and interact with all other classes together.

Upon starting the application, the view is set to contain a TextView to display the information, and a Button to start the recording. The application is then initialised by reading in the data using a DataParser object, which will be discussed later. Data is provided in the application if no file is detected, such as for the very first use. The call to analyse data is also made here, such that the SVM is already set up.

When the button is pressed, we activate all sensors at this time by creating instances of the necessary objects in the 'recorders' package. Each of the sensors collects data for one minute, before collecting the data and inputting the necessary values into the SVM.

All data results and the final decision is outputted to this Activity, where it gives the answer. However, the supervised learning element is implemented using a Dialog. This allows us the user to provide data about whether they are currently stressed. This is also recorded, and all information is added to the SVM, where it is re-analysed.

All summary data is displayed until the application is exited or another test is started.

3.4.2 com.hari.se4911.stresstester.recorders

We must implement our own SensorEventListeners. Firstly, this is done using the developed StressSensorEventListener object, an implementation of the SensorEventListener interface. This takes in two other SensorEventListener implementations, the HygrometerRecorder and AccelerometerRecorder (discussed below). This listener, which is directly used by our MainActivity, delegates its information between the two accordingly.

The AccelerometerRecorder is used to detect changes in the accelerometer. In particular, we set up three ArrayLists that will continually add substantial changes in direction in the x, y or z direction. Every time the sensor detects a change, the object will compare the current acceleration and the previous acceleration. If the change is beyond a magnitude of $2ms^{-2}$, we consider this a large change in acceleration for it to be considered a physical "turn". Each of these turns are stored in a primitive array for analysis.

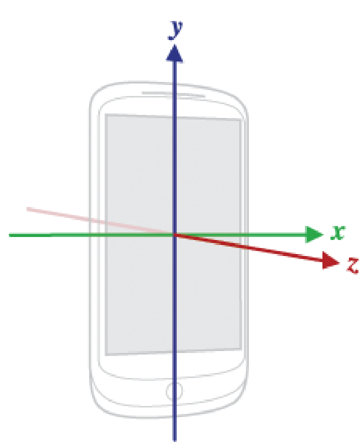

Figure 5 – Perpendicular components x, y and z, that an accelerometer can detect changes in

The HygrometerRecorder simply records humidity readings throughout the minute time period. This occurs constantly i.e. whenever the sensor can pick up data. The data returned is simply an ArrayList of the relative humidity at that point in time.

Lastly, the VoiceRecorder object is simply another Activity, and does not depend on the hardware using SensorEventListeners. The AudioRecord object is necessary for recording via microphone, which is specified in the implementation. This is done with a sample rate of 12000Hz, which is what is most effective for the Samsung Galaxy S4. During the minute, the microphone will record as much data as the object can store at a time. When this limit is filled, the amount is averaged out to find an average amplitude over this time. These averages are stored away in an ArrayList, and used later in the analysis.

3.4.3 com.hari.se4911.stresstester.results

This package contains all stored data and objects directly related to stress, that all directly help produce the output necessary for our MainActivity, let alone determine a result based on input data.

A StressResult object was created to collate all collected information from our recorders, and determine a result when combined with other classes. We determine the significance of the number of turns in this case by considering how many changes there were in acceleration in all directions. We focus on the changes in direction in the z direction, namely how many there are in comparison to the x and y directions (which in theory should be minimal).

The hygrometer readings are used to take a simple average. This is because there is no need to worry about how it changes over time and should be constant in the ideal world. The solution is based on a person’s stress level at a point in time, rather than assessing how the person’s stress levels change throughout the minute. Thus, even though it is possible to assess how respiration may change over the minute to gather more data, we do not consider this for our binary classification.

The voice readings, when taken accurately, have been deduced to exceed a level of ‘75’[8] when one’s voice is “loud”, as mentioned in 3.3.2. We take a note of how many of these values exceeds this threshold value, as a proportion of the total number of read values. Furthermore, we take a simple

[8] Units provided on phone are arbitrary

average of the numbers to help assess what type of consistency we are getting. This number is also stored away.

From here, we can instantly deduce if one is stressed using our deduced threshold values mentioned above. We can also substitute these values directly into our SVM algorithm using the DataAnalyzer class (discussed below).

Of course, these objects are stored in persistence. This is done using the CSV file, which can be used for the application at any time. We must note that the CSV file is only accurate for one user profile at a time, since the data is relevant to them.

Our DataParser class is used at the very initial stage of the application launch, when taking in our CSV file and reading the data in. This object converts our data into an ArrayList of StressResult objects. The ArrayList is returned and used for the DataAnalyzer.

The DataAnalyzer object contains the SVM logic. When created, it takes a List of StressResults and asserts what the SVM vectors are. Thus, this class is mainly an implementation of the linear SVM, which can also be used to substitute values to provide an answer, as discussed before.

Lastly, we return a different type of exception in the event the hardware fails[9]. Thus, we have created a NoResultsException, specifically designed to label at what points this occurs. This is used in many of the classes to eliminate unnecessary data.

3.4.4 UML Diagram

On the following page is Figure 6, summarising the discussed classes. We note that not all fields and methods are inserted, but a summarised version showing only certain elements is presented. This, complemented with the previous three sections, will provide us with a fundamental understanding of how the application works to produce our result.

[9] The hardware would at times stop working, and give null results.

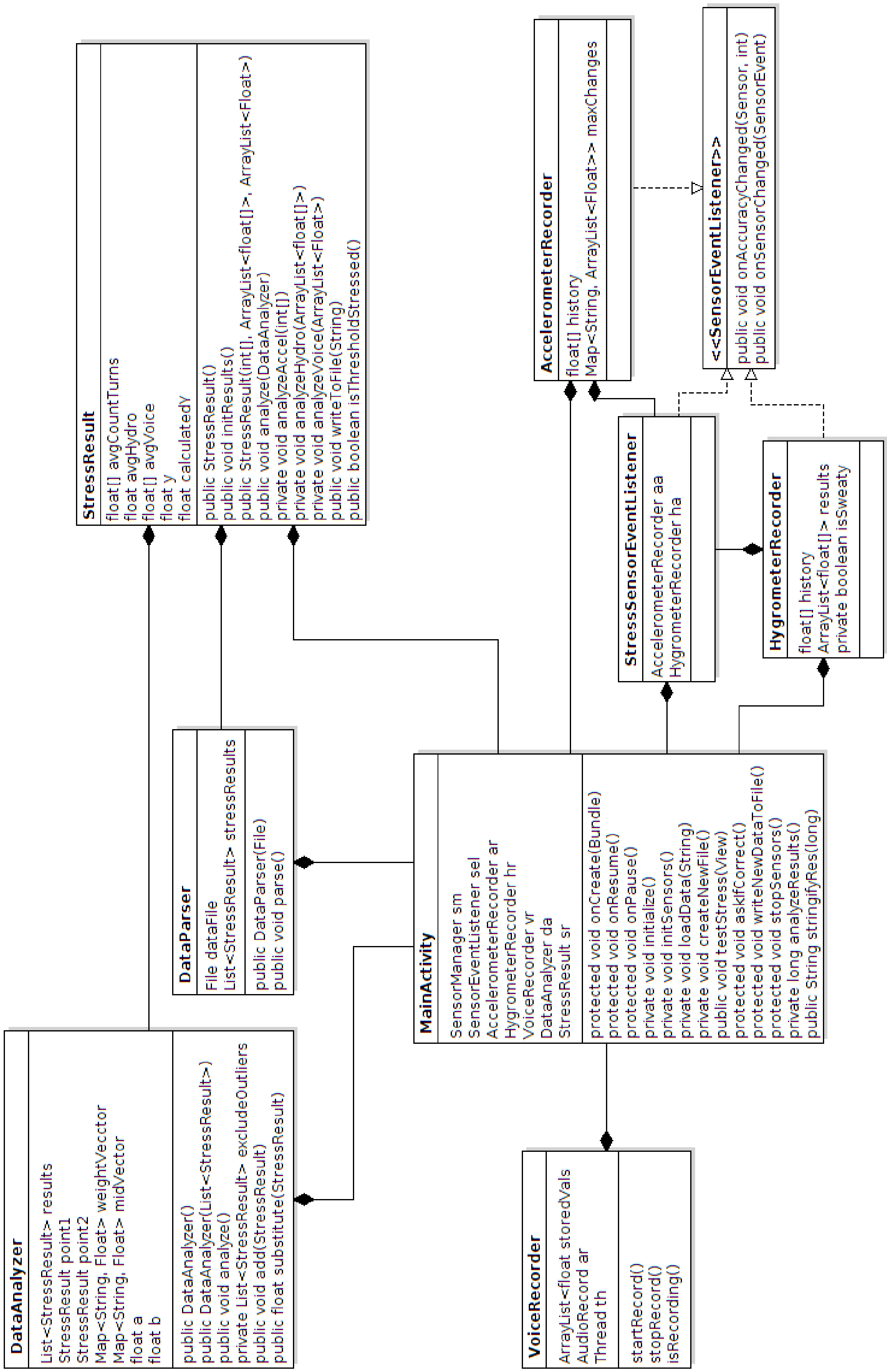

Figure 6 - UML Diagram representing implemented solution

4. EVALUATION

From here, we test our application for evaluation purposes. Evaluation is done using machine learning metrics to deduce to what degree of accuracy we have deduced one's stress level.

The following includes a testing procedure, how we have deduced accuracy and a discussion of our results.

4.1 Testing procedure

The testing procedure is very similar to what was used to test the application and acquire data. We found this was effective in understanding the data, and fell in line with our aim of deducing the accuracy of the application.

1. 2 people are taken, and kept in a calm and relaxed environment. These people are to use the application in the following steps. No other hardware apart from the device is used.
2. The first user is to start the test whilst holding the phone to their face in the manner of a phone conversation, before speaking into it for one minute, with all hardware activated. The passage "The quick brown fox jumped over the lazy dog" is repeated into the application for the minute, whilst holding the phone. This is repeated ten times.
3. This time whilst watching a scary film clip that incites shock in the user. Upon shock, the user is asked to re-use the application. This happens on 3 separate occasions.
4. All data was collected from the application using the generated CSV file, which was used in our analysis.
5. This process is repeated for the second user.

This method is repeated twice more (without notice), on different days, to ensure reliability. Thus, we have encapsulated involuntary relaxed and stressed states at instantaneous points in time.

4.2 Method of analysis

We will be measuring to what degree the application is accurate using a number of measures to classify the elements of our data set. This data set is easily extracted using the CSV file generated, as mentioned before. Figure 7 (below) presents a number of measures we will be using on the data set, specifically relevant for binary classification.

Measure	*Definition*	*Equal to*	*Estimates*
number of positives	$Pos = \sum_{x \in Te} I[c(x) = \oplus]$		
number of negatives	$Neg = \sum_{x \in Te} I[c(x) = \ominus]$	$\lvert Te \rvert - Pos$	
number of true positives	$TP = \sum_{x \in Te} I[\hat{c}(x) = c(x) = \oplus]$		
number of true negatives	$TN = \sum_{x \in Te} I[\hat{c}(x) = c(x) = \ominus]$		
number of false positives	$FP = \sum_{x \in Te} I[\hat{c}(x) = \oplus, c(x) = \ominus]$	$Neg - TN$	
number of false negatives	$FN = \sum_{x \in Te} I[\hat{c}(x) = \ominus, c(x) = \oplus]$	$Pos - TP$	
proportion of positives	$pos = \frac{1}{\lvert Te \rvert} \sum_{x \in Te} I[c(x) = \oplus]$	$Pos / \lvert Te \rvert$	$P(c(x) = \oplus)$
proportion of negatives	$neg = \frac{1}{\lvert Te \rvert} \sum_{x \in Te} I[c(x) = \ominus]$	$1 - pos$	$P(c(x) = \ominus)$
class ratio	$clr = pos / neg$	Pos / Neg	
(*) accuracy	$acc = \frac{1}{\lvert Te \rvert} \sum_{x \in Te} I[\hat{c}(x) = c(x)]$		$P(\hat{c}(x) = c(x))$
(*) error rate	$err = \frac{1}{\lvert Te \rvert} \sum_{x \in Te} I[\hat{c}(x) \neq c(x)]$	$1 - acc$	$P(\hat{c}(x) \neq c(x))$
true positive rate, sensitivity, recall	$tpr = \frac{\sum_{x \in Te} I[\hat{c}(x) = c(x) = \oplus]}{\sum_{x \in Te} I[c(x) = \oplus]}$	TP / Pos	$P(\hat{c}(x) = \oplus \mid c(x) = \oplus)$
true negative rate, specificity, negative recall	$tnr = \frac{\sum_{x \in Te} I[\hat{c}(x) = c(x) = \ominus]}{\sum_{x \in Te} I[c(x) = \ominus]}$	TN / Neg	$P(\hat{c}(x) = \ominus \mid c(x) = \ominus)$
false positive rate, false alarm rate	$fpr = \frac{\sum_{x \in Te} I[\hat{c}(x) = \oplus, c(x) = \ominus]}{\sum_{x \in Te} I[c(x) = \ominus]}$	$FP / Neg = 1 - tnr$	$P(\hat{c}(x) = \oplus \mid c(x) = \ominus)$
false negative rate	$fnr = \frac{\sum_{x \in Te} I[\hat{c}(x) = \ominus, c(x) = \oplus]}{\sum_{x \in Te} I[c(x) = \oplus]}$	$FN / Pos = 1 - tpr$	$P(\hat{c}(x) = \ominus \mid c(x) = \oplus)$
precision, confidence	$prec = \frac{\sum_{x \in Te} I[\hat{c}(x) = c(x) = \oplus]}{\sum_{x \in Te} I[\hat{c}(x) = \oplus]}$	$TP / (TP + FP)$	$P(c(x) = \oplus \mid \hat{c}(x) = \oplus)$

Figure 7 – Quantities and evaluation measures for classifiers on our test set (*Te*)[10] (Flach, 2012).

10 "Symbols starting with a capital letter denote absolute frequencies (counts), while lower-case symbols denote relative frequencies or ratios. All except those indicated with (*) are defined only for binary classification. The right-most column specifies the instance space probabilities that these relative frequencies are estimating." (Flach, 2012)

4.3 Results

Below is a summary of the calculations, as per Figure 7, as well as our comparison of expected results vs. actual results. We note that our results are split into groups of 10, with a final evaluation of all information as a whole as well. This is imperative to understand how the accuracy of the machine increases with usage. A final graph showing how accuracy increases with every use is used to further demonstrate this[11].

Sample range vs accuracy metric	**1-10**	**11-20**	**21-30**	**31-40**	**41-50**	**51-60**
number of positives	4	14	18	18	23	30
number of negatives	6	6	12	22	27	30
number of true positives	0	10	13	13	18	24
number of true negatives	6	6	12	20	22	25
number of false positives	4	4	5	5	5	6
number of false negatives	0	0	0	2	5	5
proportion of positives	40.00%	70.00%	60.00%	45.00%	46.00%	50.00%
proportion of negatives	60.00%	30.00%	40.00%	55.00%	54.00%	50.00%
class ratio	0.67	2.33	1.50	0.82	0.85	1.00
(*) accuracy	60.00%	80.00%	83.33%	82.50%	80.00%	81.67%
(*) error rate	40.00%	20.00%	16.67%	17.50%	20.00%	18.33%
true positive rate, sensitivity, recall	0.00	0.71	0.72	0.72	0.78	0.80
true negative rate, specificity, negative recall	1.00	1.00	1.00	0.91	0.81	0.83
false positive rate, false alarm rate	1.00	0.29	0.28	0.28	0.22	0.20
false negative rate	0.00	0.00	0.00	0.09	0.19	0.17
precision, confidence	0.60	0.60	0.71	0.80	0.81	0.81

Table 4 – Results from the 'StressTester' application

[11] Collected user data is taken as an unweighted average for this analysis – the users are considered equally important sources of information.

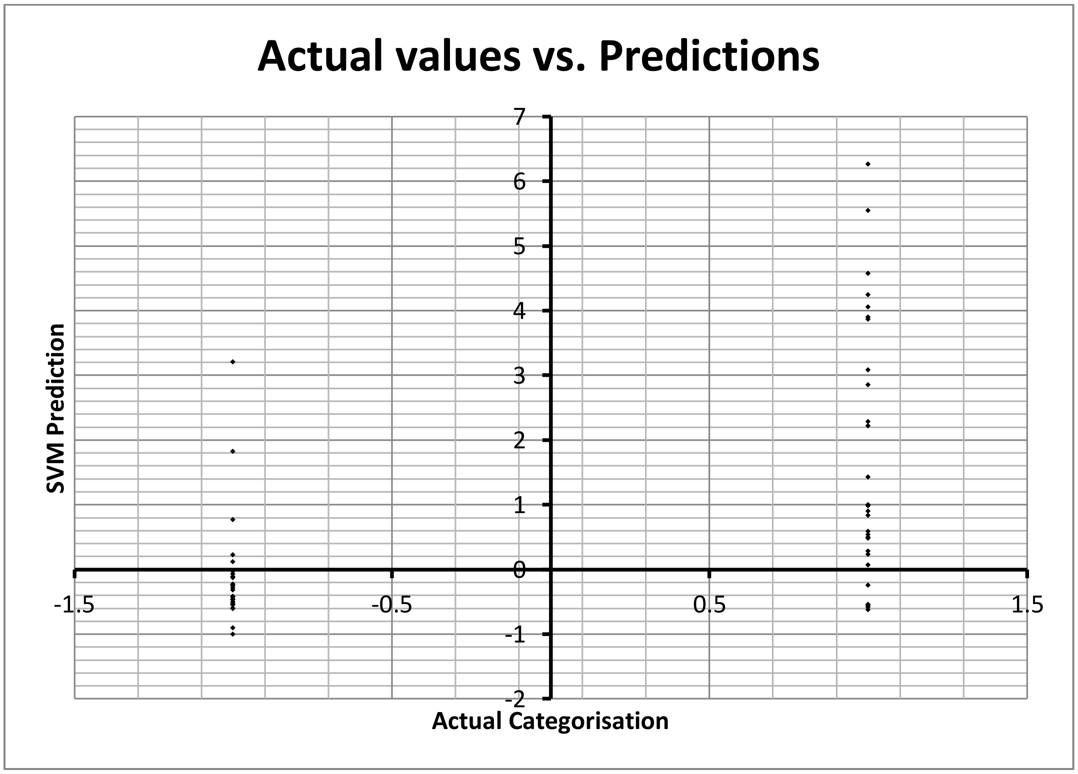

Figure 8 - a graphical representation of the raw data, indicating location of accurate and inaccurate data[12]

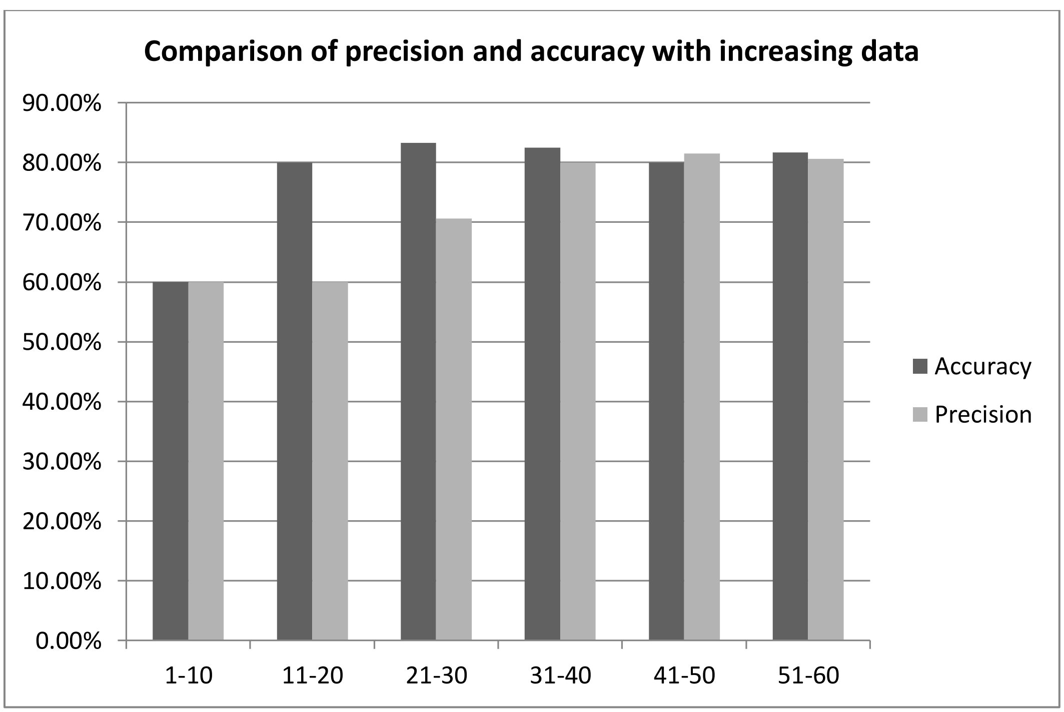

Figure 9 - Graphical analysis of consistent accuracy with gaining confidence

[12] Data in the top-right and bottom-left quadrants are accurate, whilst the other two result in an incorrect classification

4.4 Discussion

Our methodology in starting this application has proven to be valid. We have first started with appropriate research, before forming hypotheses and a solution. Our solution is then tested against the hypothesis using validated methods. These methods can be considered valid as all can be justified with relevance to the aim.

Thus, we have validly deduced the accuracy of our solution ranges from 60.00 to 83.33%. We also notice the trend of an increase in accuracy as more results are gathered past the test data. Our hypothesis that lies directly in line with the basic principles of machine learning hold true: the accuracy of the system increases upon further use. In particular, we deduce that the machine learns about the stress behaviours of the single user at any point in time.

Our accuracy can be considered relatively low in terms of practical use. Furthermore, whilst the results were of some consistency, it pales in comparison to that of current contemporary solutions and research. Reasoning behind these trends can be attributed to a number of factors.

The hardware used proved to be relatively inaccurate from our experience, with the microphone in particular. This appeared to produce spikes in listening and, at times, did not detect any noise. Due to this inconsistency, some results may appear to be largely skewed to high values, as seen in Figure 8. We also note that there was a lower range for negative values over positive values obtained from the SVM. This can also be attributed to random biases – despite ultimately providing a unbiased number of decisions of each classification.

We also recognise the importance of which data to collect. In comparison to a lot of other state-of-the-art solutions mentioned above, the accuracy achieved is on par. Yet the full potential of the application has still not been achieved due to hardware constraints.

There is a large variety of data which we can collect to detect stress. One of those most important ones is the heart rate, which did not receive any attention in our current progressive solution. We do recognise from our previous literature review that heart rate is one of the most important factors to consider and, thus, our application is inferior to others based on this simple principle.

However, that is not to say that the solution does not have potential. A listener for the heart rate sensor can still be implemented and used with the correct hardware provisions. Once implemented, we will be sure to see a much higher accuracy from the start, and a steeper climb in accuracy through its use.

We then have the question of our relatively high accuracy in comparison to other state-of-the-art technology, despite the flaws aforementioned. In comparison, we note that we have used more commonly associated physical stress symptoms and stayed further away from the physiological symptoms. As also mentioned before, physical symptoms are a primary source of the solution that provided the highest degree of accuracy (91 – 95%) (Sharma & Gedeon, 2013). These physical symptoms mainly include pacing. Furthermore, only strong physiological symptoms were used for data collection, since there is a strong disparity between stressed and not stressed.

Lastly, we acknowledge where technology has currently progressed to, and how it will increase with time. We are dealing with hardware that is not necessary for the primary functions of the phone. Errors produced from most hardware were dealt with effectively by smoothening out the averages. However, as time progresses, technology will progress. We can expect that the future of the application, and other solutions for that matter, is promising with an increasing level of precision and accuracy that comes with the bettered hardware.

We conclude our discussion with recognising that despite the previously mentioned flaws in the solution, it is also innovative in its approach to detect stress. Whilst this is not the complete answer, it is a piece of the puzzle to the ultimate solution. This solution is by no means astray from the path of being a state-of-the-art application for its purpose – we see that it grows strongly in precision as a user uses it more, which is highly important.

5. CONCLUSION

5.1 Future work

The potential for this application, upon future work, has the potential to be the current state-of-the-art technology. We also recognise its limits to Android technology, and other uncompleted work suggested in the proposal. In particular, however, primary future work specifically refers to the following:

5.1.1 Use of heart rate sensor

The state-of-the-art heart rate sensor is property of Samsung, and currently is only available on Samsung devices. In the future, we hope to see this sensor be used more prominently. Future work currently tends towards the use of external hardware, such as electronic wrist bands that communicate with the device wirelessly – namely, using Bluetooth technology.

This has potential to be used in future work to provide a much more accurate result, as it is a key stress response to consider, as well as its progression to ultimately be implemented into phone technology.

5.1.2 Stable hardware

Currently even the most basic of hardware, such as the microphone, still has errors. Of course, these errors can be eliminated as we have so done. However, more work must be done in order to completely eradicate this error. An example of a potential avenue is through Kernel Density Estimation (Duong, 2001), which is one of the most popular methods of reducing variances and error due to outliers.

Nonetheless, with future work aside, technology is still yet to evolve with time to produce more stable hardware that is less prone to errors. Thus, the application must be adapted to this future hardware such as due to changes in the Android API.

5.1.3 Furthered machine learning algorithms

Whilst a support vector machine is of interest, we should also consider other machine learning algorithms, as their potential in this application has not been seen by the vast majority of research. Rather, we have seen most have also created a linear SVM such as in our current solution. Discussed earlier in the Section 3.1.1 was the idea of using Sequential Forward Selection (SFS) or a random forest decision tree.

These two in particular are advanced and popular methods of binary classification. The SFS algorithm in particular has the potential to obtain the importance of certain features. SVMs, by nature, do not[13]. Thus, we see potential for a faster application. Whilst this is not currently relevant due to the small amount of data, this will prove more relevant in the future with more data. Furthermore, it will be relevant when data is inevitably skewed, as we have learnt that biased data does not react well with an SVM.

Random forest decision trees are also renowned to be fast in comparison to the SVM. This involves a combination of "tree bagging" and the random subspace method (a random choice of subsets to work it) throughout data samples. Essentially, the result is the use of many weak hypotheses from smaller data sets to form a hypothesis of strong certainty via ensemble theory (Rokach, 2010)– this is referred to as ensemble learning.

Despite the speed of the random forest decision tree, it actually requires more brute force to achieve the same accuracy as an SVM. This is despite the fact that it is very good at handling a large amount of data. Nonetheless, this is a general principle, and all machine learning algorithms have different effects depending on the data set used. Thus, we encourage this to be used in the future too.

Whilst we recognize the potential for other methods of classification, such as the suggested Gaussian Model Machine for non-binary classification, these are not of primary concern for reasons stated above. We choose to stick with the most effective binary classification to achieve our aim.

5.1.4 Integration into natural use

Whilst the purpose of this thesis was in the name of research, we recognise the benefits of having a practical application. Thus we look for a suggestion regarding how this application can be used naturally rather than on a conscious level.

One particular suggestion includes running the application in the background and having it activated or deactivated based on whether the user is on a phone call or not respectively. This allows the user to hold the device in the required manner whilst being on the phone call without too much discomfort or conscious thought. This, in particular, is not only practical, but will also show a better reflection on the accuracy of the app if tested in that fashion over a long period of time.

[13] Whilst SVMs take the two closest distances to anchor upon, and thus ultimately works on brute force to acquire accuracy (Weston, et al., n.d.)

5.1.5 Persistence

We have mentioned before that current phones are capable of global communication using data services. Thus, a persistence model can be implemented to send the data to a central collated source. This has not been done in the past, but can add a lot of value. It would help create a stronger data set, since stress varies from person to person. Ultimately, this can help create profiles for people, and lead to a more effective application.

A recommended method of doing this would be through the use of a RESTful service, with a database set up. Our StressResult object can be extended to include a user profile. This would not require a data-intensive request or response, relative to the amount of value that could be gained. Thus, this is recommended for furthering our current solution

5.2 Final words

To summarise our work, our research has proven the use of four key symptoms to detect stress – one's heart rate, use of voice, sweatiness of palms, and erratic movement (namely pacing up and down over a small area). The most relevant analysis is using machine learning. We choose binary classification due to its relevance to the people, and the large disparity between a calm state and one that is dangerously distressed.

We proceed to develop the application by first collecting raw data and evaluating its integrity in helping deduce the onset of stress. Once confirmed, an application was developed to integrate all data together, combined with machine learning algorithms to further categorise the data.

From our evaluation, it is very clear that it is a stable and robust application that serves its purpose as a solution to our aim. Whilst there are slight aesthetic tweaks to be made[14], there is still a large potential for the system to grow and achieve accuracies beyond comparison, with the current accuracies ranging from 60.00 to 81.67%. However, with future work, we can guarantee higher levels of accuracy to achieve or surpass the standards of current state-of-the-art technology.

[14] The application would at times reset due to changes in orientation.

6. BIBLIOGRAPHY

Abraham, S. et al., 2011. *Green Fluorescent Protein-Based Biosensor To Detect and Quantify Stress Responses Induced by DNA-Degrading Colicins,* Woolongong, NSW, Australia: American Society for Microbiology.

American Psychology Association, 2014. *Stress The different kinds of stress.* [Online]
Available at: http://www.apa.org/helpcenter/stress-kinds.aspx
[Accessed 20 May 2014].

Azumio, 2012. *Stress Check | Azumio.* [Online]
Available at: http://www.azumio.com/apps/stress-check/
[Accessed 3 April 2014].

Bousefsaf, F., Maaoui, C. & Pruski, A., 2013. *Remote assessment of the Heart Rate Variability to detect mental stress,* Metz, France: Laboratoire de Conception, Optimisation et Modelisation des Systemes (LCOMS).

CalmClinic, 2014. *How Walking Works to Cure Anxiety « Calm Clinic.* [Online]
Available at: http://www.calmclinic.com/anxiety/treatment/walking-works
[Accessed 22 Sep 2014].

Carlson, N. R., 2007. *Physiology of Behavior.* Amherst, USA: Pearson.

Chandrasekeran, V., 2010. *Measuring Vital Signs Using Smart Phones,* Denton, TX, USA: University of North Texas.

Chang, K.-h., Fisher, D. & Canny, J., 2011. *AMMON: A Speech Analysis Library for Analyzing Affect, Stress, and Mental Health on Mobile Phones,* Berkley, CA, USA: University of California at Berkeley.

Choi, J. & Gutierrez-Osuna, R., 2009. *Using Heart Rate Monitors to Detect Mental Stress,* Texas: Texas A&M University.

Choi, J. & Gutierrez-Osuna, R., 2010. *Estimating Mental Stress Using a Wearable Cardio-Respiratory Sensor,* Texas: Texas A&M University.

De Choudhury, M., Gamon, M., Hoff, A. & Roseway, A., 2013. *"Moon Phrases": A Social Media Faciliated Tool for Emotional Reflection and Wellness,* Redmond WA, USA: Microsoft Research.

Dey, A. K. & Abowd, G. D., 1999. *Context-Awareness, Towards a Better Understanding of Context and Context Awareness,* Atlanta, GA, USA: Graphics, Visualization and Usability Center and College of Computing, Georgia Institute of Technology.

Duong, T., 2001. *An introduction to kernel density estimation.* [Online]
Available at: http://www.mvstat.net/tduong/research/seminars/seminar-2001-05/
[Accessed 24 Oct 2014].

Education Portal, 2003-2014. *What is Humidity? - Definition, Measurements & Effects Video - Lesson and Example | Education Portal.* [Online]
Available at: http://education-portal.com/academy/lesson/what-is-humidity-definition-measurements-effects.html#lesson
[Accessed 12 Oct 2014].

Ertin, E. et al., 2011. *AutoSense: Unobtrusively Wearable Sensor Suite for Inferring the Onset, Causality, and Consequences of Stress in the Field Onset, Causality, and Consequences of Stress in the Field,* Seattle, WA, USA: SenSys.

Flach, P., 2012. *Machine Learning: The Art and Science of Algorithms that Make Sense of Data.* Cambridge, CB2 8RU, UK: MPG Books Group.

Google, 2014. *Environment Sensors | Android Developers.* [Online]
Available at: http://developer.android.com/guide/topics/sensors/sensors_environment.html
[Accessed 15 Apr 2014].

Google, 2014. *Motion Sensors | Android Developers.* [Online]
Available at: http://developer.android.com/guide/topics/sensors/sensors_motion.html
[Accessed 15 Apr 2014].

GSMArena, 2014. *Samsung Galaxy S5 - Full phone specifications.* [Online]
Available at: http://www.gsmarena.com/samsung_galaxy_s5-6033.php
[Accessed 10 Aug 2014].

Hernandez, J., Drevo, W., Picard, R. W. & Hoque, M. E., 2012. *Mood Meter: Counting Smiles in the Wild,* Cambridge, MA, USA: UbiComp.

HowStuffWorks, 2009. *HowStuffWorks Hygrometer.* [Online]
Available at: http://science.howstuffworks.com/nature/climate-weather/meteorological-instruments/hygrometer-info.htm
[Accessed 15 April 2014].

How, T.-V., Chee, J., Wan, E. & Mihailidis, A., 2013. *MyWalk: A Mobile App for Gait Asymmetry Rehabilitation in the Community,* Toronto, Canada: University of Toronto.

iHealth, 2012. *iHealth Wireless Blood Pressure Wrist Monitor Mobile Health Products.* [Online]
Available at: http://www.ihealthlabs.com/wireless-blood-pressure-monitor-feature_32.htm
[Accessed 23 May 2014].

Küblbeck, C. & Ernst, A., 2006. Face detection and tracking in video sequences using the modified census transformation. *Journal of Image Vision Computing,* 24(6), p. 9.

Lee, Y. et al., 2013. *SocioPhone: Everyday Face-to-Face Interaction Monitoring Platform Using Multi-Phone Sensor Fusion,* Taipei, Taiwan: MobiSys.

LiKamWa, R., Liu, Y., Lane, N. D. & Zhong, L., 2013. *MoodScope: Building a Mood Sensor from Smartphone Usage Patterns,* Beijing, China: MobiSys.

Logan, A. G. et al., 2007. Mobile Phone–Based Remote Patient Monitoring System for Management of Hypertension in Diabetic Patients. *American Journal of Hypertension, Ltd,* 2007(20), p. 7.

Lu, H. et al., 2009. *SoundSense: scalable sound sensing for people-centric applications on mobile phones,* New York: ACM.

Lu, H. et al., 2012. *StressSense: Detecting Stress in Unconstrained Acoustic Environments using Smartphones,* Pittsburgh, USA: UbiComp.

National Institutes of Health, 2014. *Stress and Anxiety: MedlinePlus Medical Encyclopedia.* [Online]
Available at: http://www.nlm.nih.gov/medlineplus/ency/article/003211.htm
[Accessed 16 6 2011].

Rokach, L., 2010. Ensemble-based classifiers. *Artificial Intelligence Review,* 33(1-2), pp. 1-39.

Samsung, 2014. *SAMSUNG Developers.* [Online]
Available at: http://developer.samsung.com/s-health-sdk#pg
[Accessed 11 Aug 2014].

Segal, R., Smith, M. & Segal, J., 2014. *Stress Symptoms, Signs &Causes: Effects of Stress Overload.* [Online]
Available at: http://www.helpguide.org/mental/stress_signs.htm
[Accessed 12 May 2014].

Sharma, N. & Gedeon, T., 2013. *Modeling Stress Recognition in Typical Virtual Environments,* Canberra, Australia: Australian National University.

Stafford, P., 2013. *Australians go local with smartphones – and three other trends from Google's latest research.* [Online]
Available at: http://www.smartcompany.com.au/technology/online/32768-australians-go-local-with-smartphones----and-three-other-trends-from-google-s-latest-research.html
[Accessed 7 May 2014].

Volterra, V., 1959. *Theory of Functionals and of Integrals and Integro-Differential Equations,* NY, USA: Dover Publications.

WebMD, 2013. *Stress Symptoms Effects of Stress on the Body.* [Online]
Available at: http://www.webmd.com/balance/stress-management/stress-symptoms-effects_of-stress-on-the-body
[Accessed 23 May 2014].

Wei, Q. & Dunbrack Jr, R. L., 2013. *The Role of Balanced Training and Testing Data Sets for Binary Classifiers in Bioinformatics,* s.l.: Plos One.

Weston, J. et al., n.d. *Feature Selection for SVMs,* Red Bank, USA: AT&T Research Laboratories.

APPENDIX

A.1 Screenshots of application

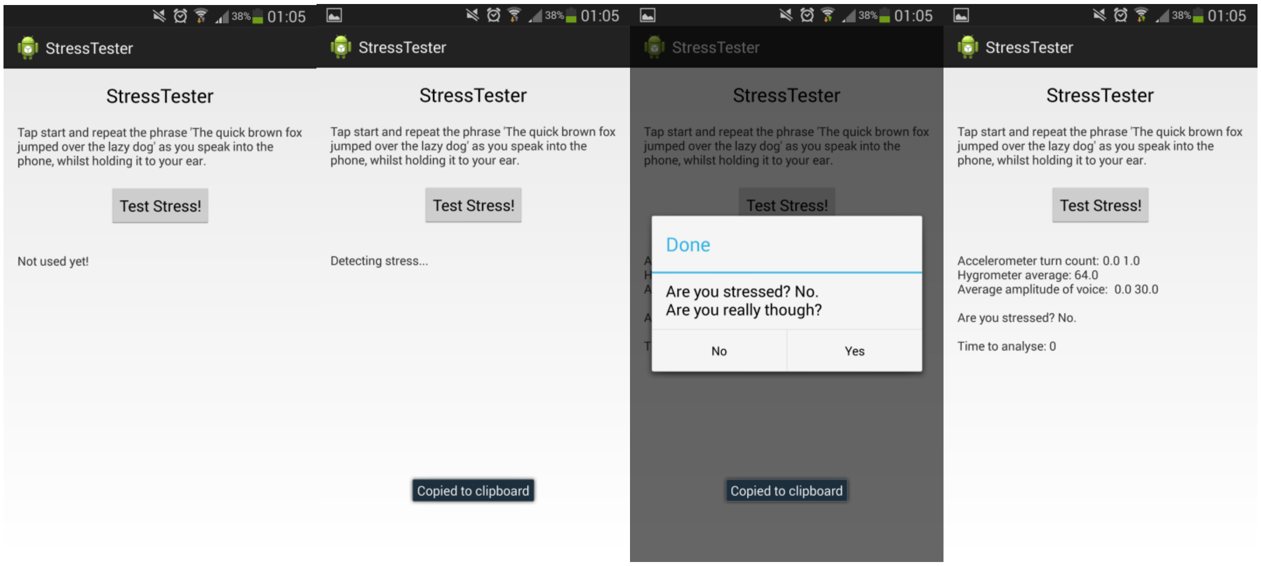

Figure 10 - A step-by-step view (from left to right) of the application

Clearly, we can see how the application goes through its sequence of events from each of the screen shots.

- The first picture is the initial state.
- The second picture is after activating the application via the "Test Stress!" button
- After one minute, the application has collected all data and is assessing the onset of stress, as per the third picture. Meanwhile, the application produces a dialog asking whether the person is stressed as a supervised learning element
- After responding, the final screen shows the raw data (purely for research purposes), and all other processes are finished off. The application lifecycle is ready to be restarted again with another click of the button.

BLANK PAGE