93601

936011

S

TOP SCHOLAR

NZQA

NEW ZEALAND QUALIFICATIONS AUTHORITY
MANA TOHU MĀTAURANGA O AOTEAROA

QUALIFY FOR THE FUTURE WORLD
KIA NOHO TAKATŪ KI TŌ ĀMUA AO!

Scholarship 2018
Technology

Abstract

Food is one of the most fundamental cornerstones in sustaining human life with its consumption intricately linked to our health. Vital as it is to our lives, properly harnessing an understanding of how food affects us offers new perspectives on topical challenges in our society requiring solutions which are both robust and dynamic to the future and ourselves. This paper explores the analysis, synthesis and integration of technological experiences in the development, prototyping, testing, evaluation and reflection of several technical outcomes, and, proposes an artificial intelligent solution aimed at addressing rising health problems stemming from nutrition misinformation.

Abbreviations

- NN: Neural Network
- CNN: Convolutional Neural Network
- AI: Artificial Intelligence
- UI: User Interface
- ML: Machine Learning
- NFA: Normalised Float Array
- CEL: Cross Entropy Loss

Table of Contents

Background

Nutrition misinformation is now the second leading global risk factor contributing towards human mortality rates[1]. Consequently, at least 2.8 million people each year die from food related health problems[2]. The growing body of medical knowledge and strong correlation between diet and personal health is causing many consumers to take nutrition decisions into their own hands. As a result individuals are becoming more reliant on websites, advertisements or social media to base their decisions; thereby furthering opportunities for misinformation. It's clear public health is at a crossroads.

Misinformed eating is a common trend shared among the world's countries and the economically impoverished are often the worst affected. While cheap food is frequently viewed as a more economical option, in the long-term, these people are ultimately robbed of the quality nutrition they need. Through misinformation from advertisements and socio-cultural factors yet to be explored, the Pan American Health Organization[3] has observed food related diseases such as Obesity and Type 2 Diabetes are is increasingly being seen as features of the poor.

Informing and educating people about their eating trends is often still the first step to managing this pandemic, thus continuing the philosophy of the World Health Organization in *the benefits of medical and related knowledge is essential to the fullest attainment of public health*[4].

This idea of education also ties into what I believe is one of the most important tools we have in managing nutrition misinformation and in turn represents a paradigm shift in modern technological trends: *the move toward adaptable and intelligent systems*. The criticality of this shift is integral to tackling such a global issue because it brings the potential for dynamic problems, such as nutrition misinformation, which have previously been considered far too complex to be understood by a computer to be solved quickly and autonomously, enabling solutions to be applied at scale.

However, meeting the infinitely variable health needs for every person globally means they must first be understood. By designing intelligent systems to interpret these dynamic problems for the first time in history we are able to emulate aspects of human capability in automated problem solving. In this way I feel that the most important distinction between traditional and artificially intelligent methods to problem solving is their ability to be applied at scale while maintaining efficiency and accuracy.

It is just as important to recognise this technological shift as it is in viewing food related health detriments as a global pandemic. An increasing number of studies have shown food related health complications are a contributing factor in pre exposing a person to various systemic diseases[5], most notably to obesity. Shown by a 2008 study[6] 35% of adults aged 20+ were overweight and had a BMI ≥ 25 kg/m^2. The worldwide prevalence of obesity has nearly doubled between 1980 and 2008 and in 2017 an estimated 205 million men and 297 million women over the age of 20 were obese. This figure encompasses more than half a billion adults worldwide and expotentially increases the risks of Coronary Heart Disease, Ischemic Stroke, Type Two Diabetes, and Cancer.. In 2015 750,000 men and women (Fig-1) were approximated to have a 90% higher chance of contracting some form of cancer who were 40% heavier than the median BMI average of the population at the time[5].

[1] "Nutritional misinformation of children: A ... - Science Direct." https://www.sciencedirect.com/science/article/pii/0193397381900149. Accessed 9 Oct. 2018.

[2] "Science Direct." https://www.sciencedirect.com/science/article/pii/0193397381900194. Accessed 8 Oct. 2018.

[3] "Home - Pan American Health Organization - PAHO." https://www.paho.org/hq/. Accessed 14 Apr. 2018.

[4] "WHO | Constitution of WHO: principles ...." http://www.who.int/about/mission/en/. Accessed 27 July. 2018.

[5] "Obesity Macula Center." 9 Aug. 2012, https://maculacenter.com/eye-news-tampa-bay/obesity-and-vision/. Accessed 11 May. 2018.

[6] "WHO | Obesity." http://www.who.int/gho/ncd/risk_factors/obesity_text/en/. Accessed 28 July. 2018.

Types of cancers associated with an increase of BMI.
(Note: BMI is defined as a person's weight in kilograms divided by the square of the height in meters) (bmi = m / h^2)
Fig 1 | Source: https://mail.bfresh.us/2011/v16/af/3810/3810.pdf

Cancer	RR (95% CI) for Men	RR (95% CI) for Women
Oesophageal adenocarcinoma	1.52 (1.33-1.74)	1.51 (1.31–1.74)
Thyroid	1.33 (1.04–1.70)	1.14 (1.06–1.23)
Colon	1.24 (1.20–1.28)	1.09 (1.05–1.13)[1]
Renal	1.24 (1.15–1.34)	1.34 (1.25–1.43)
Endometrium	NA	1.59 (1.50–1.68)
Gallbladder	1.09 (0.99–1.21)[1]	1.59 (1.02–2.47)

Inherent with the ensuing challenges that come with a globally established issue these preliminary factors point to the need for future proof and robust solutions. Moreover, global establishment demands scalability as different people have different requirements, especially given my emphasis placed on education. Understanding the socio-cultural circumstances and economic factors which also play a major role in the prevalence and severity of this issue must also be considered and researched. These broad factors will contribute to the metric by which I will evaluate fitness for purpose throughout my report.

Issue

Current techniques to manage food related health problems stemming from nutrition misinformation require information be gathered from an external source, yet the the availability of this sort of information is limited. The American Oxford Journal[7] suggests the most productive means comes from medical professionals such as Nutritionists or Dietitians. However, this raises an important social and economic consideration as often these methods prove highly expensive, with the most recent comprehensive study by *One Medical*[8] finding nutrition counseling running as high as $125 USD an hour. It's clear any solution must pertain to the requirements of the economically disadvantaged in my society, as previously explored these are often the most needy.

As an alternative, freely available sources of information are accessible to people, yet, are often comparatively limited or misinformative proving too inconvenient to read in real life. To cope numerous innovations have been made in an attempt to bridge this gap but are consistently hindered by the inherent complexity of the issue and diversity of their target market, preventing solutions being applied at scale.

Existing Practice

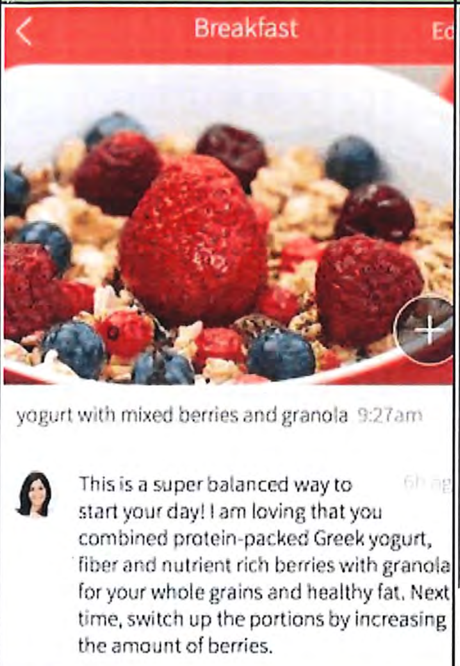
Fitness trackers such as iMacro (Fig-3) attempt assimilate a tailored nutritionist treatment through a tedious reliance on self data entry coupled with basic automation mechanisms to emulate human feedback. I, however, found this process to be very time consuming with each meal talking around 1 - 2 minutes to record, proving the accommodation of detailed feedback is at the expense of efficiency. A manual nature also leads to frequent errors in consumer self reporting, which, Cornwall University attributes to typically have a error rate exceeding 400 calories per day[9]. This uninformed practice does not invoke emotional resonance since over the course of days to weeks this systematic error is amplified leading to misguided diet changes.

[7] "Long-term weight loss maintenance | The American ... - Oxford Journals." 1 Jul. 2005, https://academic.oup.com/ajcn/article/82/1/222S/4863393. Accessed 21 May. 2018.

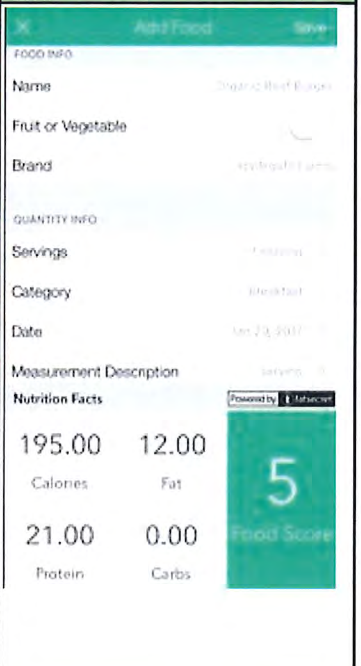
[8] "One Medical." https://www.onemedical.com/. Accessed 9 Oct. 2018.

[9] 27 Jul. 2017, https://arxiv.org/abs/1707.08816. Accessed 15 May. 2018.

Rise mobile nutritionist support.
Fig 2 | https://www.rise.us/



ImacroNutrition Self Reporting App
Fig 3|http://www.imacroapp.com/



Solutions that attempt to address these issues do so with varying levels of success. A technique commonly used to reduce human reliance is making use of a device camera. *Rise* (Fig-2*)* transplants real world nutritionists into a smartphone app so they can conduct analysis on a per client basis real time. Yet, use cases are again slow. It's clear that the mere transposing of medical professionals into a digital medium is far from a workable solution. Furthermore, the expensive subscribable nature reintroduces a hindrance to the economically disadvantaged, and, also lacking the ability to upscale prevents it from helping a large number of people. These problems are critical to overcome in tackling my identified issue of widespread nutrition misinformation.

Issue In Context

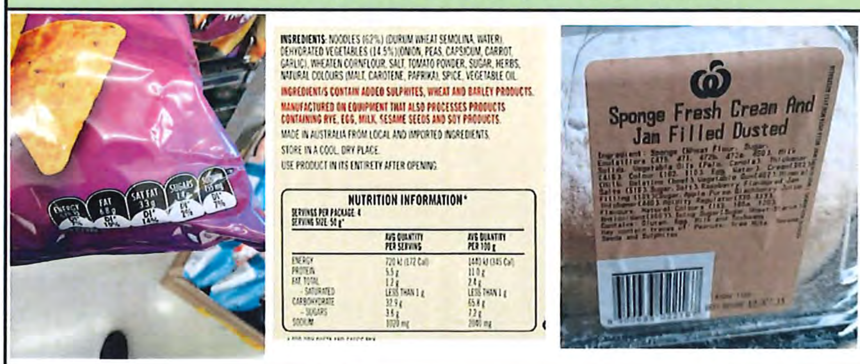
Pre-packaged meals are another important context of this issue. As I frequently help out with the family shopping I notice this to be especially prevalent within supermarkets (Fig 4a,b,5). There are several ways this nutrition information is communicated. The first is the Health Star Rating. This is a 5 point score designed to give consumers a quick indication of the '*healthiness*' of a meal. Minimal text and a quickly scannable visual aesthetic mean this practice is able to convey meaning efficiently and also be understood by linguistic and numerically illiterate people. However, since it is not required by New Zealand law to be shown on products its often omitted from unhealthy items such as the Dorito packet of chips (Fig-5)

Disproportionate Health Star Ratings
Fig 4 (Left: fig 4a, Right fig 4b)| Source: Self

Disproportional shading of the health star rating is common and detrimental to it's coherent design. In Fig-4 both products (*capers* and *vegetable oil*) have roughly the same numerical rating but a drastically different number of stars shaded. This inhibits consumers from being able to visually scan products and goes someway to furthering nutrition misinformation.

Another relevant practice are conventional summary tables. Here detailed information is provided by law so it can be counted to be present.

Nutritional Summaries from my local supermarket | Fig 5 Source: Self



However, I have found quantifying numbers and percentages used to represent ingredients to be ambiguous and hard to relate back to a specific meal instance. Furthermore, all summaries provide feedback through either percentages or grams, posing a barrier to mathematically illiterate or people who use the imperial system.

The small space which summaries are allocated on packaging means character height is often reduced to a minimum. Moreover, the way some packaging bends and folds mean text can easily get hidden in crevices. Often it is in the economic interests of food companies to make these summaries hard to read as they contain information which might reveal the true nature of the product and discourage consumers from purchasing that item. For example, how could the everyday person understand how *butylated hydroxyanisole* could affect their body?

Fig 6	Source: self	*Menu Board with calorie count*

It was has here I began to notice another key context related issue especially prevalent in unique meal cases from cafes and restaurants. These almost always require a customised nutritional analysis be made due to the unique nature of their food, hence the aforementioned practices would quickly become obsolete. An attempt to mitigate these factors can be seen at a Chipotle's restaurant while I was in the USA. Writing the numbers of calories next to each meal item provides a quantitative understanding of energy content. However it assumes customers have an understanding of how calories affect health. Ignoring my research knowledge I felt I was being led to believe that more calories are less healthy, when in fact, extrapolating from a discussion with Danijela (my key stakeholder discussed below) showed the importance in differentiating the benefits of complex carbs in sustaining energy and simple carbs for short-term energy.

Furthermore, this particular store serves food which is inherently variable in nature, *ie burritos with customisable fillings and toppings*, so a high degree of inaccuracy is present between what you get and the calorie count on the board. This process fundamentally lacks in its dynamic ability to adapt around different food groups, proving a "*one size fits all*" approach in terms of analysis is not functional in both a social and physical sense.

Moreover, the process itself does not inform customers as to other nutritional properties such as protein, vitamins or minerals additional information could be provided upon request. However, during my visit I sensed there was a rush to serve as many people as fast as possible so as a social consideration people could feel pressured to make their decisions very fast, eliminating the window to consult staff about left out nutritional information.

In response to this, any solution I develop should function quickly to fit seamlessly into this customer employee dynamic and time-pressured contexts. Efficiency will therefore also become a fundamental physical specification in my brief and wider solution. In either case to facilitate these values in response to the changeable nature of food a high degree of autonomy is desirable because it has the potential to both lower the cost of nutrition analysis and meet the tailored and diverse nature solutions which global scalability demands. Autonomy will, therefore, continue to be a pivotal aspect to consider in my brief.

Stakeholder Analysis

As a deeply embedded problem in many societies around the world, any solution orientated at tackling nutrition misinformation will impact a lot of people. Considering this I must respond appropriately and conduct my research inclusively of these people to be affected. To achieve this I have selected a key, wider and technical stakeholder. Given these people's heavy involvement in my solution they will come to represent a large portion of the wider physical and social nature of my report hence influence how fitness for purpose will be evaluated.

Nutritionists

To have any hope at making a positive contribution against nutrition misinformation I must first understand existing techniques in play to manage it. Nutritionists are a key stakeholder group to consider in this sense. I contacted a local nutritionist in my area called *Danijela Unkovich* from *Feelfresh Nutrition*[10] who, during an initial interview, agreed to her role as my primary stakeholder. A key attribute of this stakeholder group is their personalised treatment as often this is what Danijela said makes consultations so effective. However, the one on one nature can significantly hinder the scalability and economic sustainability of these techniques when applied at scale, relevant to my wider stakeholders. It is therefore essential that my project considers this to avoid merely transplanting already used techniques into a digital solution, I must extrapolate and innovate.

The importance of this consideration was unearthed during an initial stakeholder interview. The primary technique Danijela said she used to help her clients was showing them how to assess their food independently. She put a particular analysis on doing so without observing nutritional summaries since, as previously discussed, they were hard for everyday people to understand and can not be relied on in every context. Danijela said this *'training'* helped her clients maintain their improved diets well into the future after they were out of her care, furthering the importance of education in managing this issue.

Experiment showing how a professional nutritionist analyses meal healthiness
Fig 7 (Top: fig 5a, bottom fig 5b)| Source: Self

Images that were presented to Danijell in random order

The order images were sored into

I wanted to better understand this technique to further inform my ideas in developing my own solution. To illustrate this I brought along five different pictures of meals ranging from unhealthy fried foods such as McDonald's fries to healthy meals such as salad (fig-7a). I asked Danijela to order these foods by how healthy she thought appropriate and by doing so revealing her visual food assessment method. Fig-7b shows how Daniela ordered salads at one end of the spectrum while cookies and Mcdonalds were at the other. What was interesting was the way Danijela explained her decision, saying different foods were assessed visually by listing out the separate parts/ingredients and then drawing on her past knowledge, determining each's overall 'healthiness'. In context, she said the salad had individual ingredients like tomatoes and lettuce which she knew to be healthy whilst the Mcdonalds burger had lots of oil and plenty of salt. For the unhealthy foods, Danijela explained these had high sugar and fat content which can cause inflammation within the digestive system. I will refer to this technique as an object based approach to meal analysis.

Abstracting from this discussion, object based analysis fundamentally shows usability as an important consideration to fit into the physical environment. Where clients were helped through the often daunting process by Danijela my solution must take complicated situations such as these and provide analysis automatically, in a usable, friendly manner bearing similarities in accuracy. This analysis of food must be both intuitive and approachable to make it usable by everyday people. This idea of user-friendliness will, therefore, play another major role in development and when evaluating the fitness for purpose of my prototypes.

It was also important to discuss the diverse contexts that my solution will be required to function within. On this topic, I spoke about my recent trip to the supermarket regarding nutritional summaries,

---

[10] "Feel Fresh Nutrition." https://feelfreshnutrition.com/. Accessed 27 Feb. 2018.

through Danijela could relate to in terms of challenges faced in nutritional understanding. However, she was quick to point out the widespread prevalence that misinformative eating trends extended to in contexts outside the ones I had already explored, suggesting I look into places such as home cooked meals, fast food outlets, drive-thrus or cafes. Abstracting, I saw in these places the employee often becomes the primary source of nutritional information and these changes to the product-customer-employee dynamic become an important social consideration. In a supermarket, a client can walk around with relative ease in contrast to a fast food drive throughs which restrict free movement.

There is also an increased element of time pressure between contexts. This made me realise designing to a rigid use case can limit the fitness for purpose and effective functionality across a diversity of contexts.

People struggling from Food Related Health Problems

People struggling with food-related health problems will make up my wider stakeholder group. In 2017 Census NZ classified this group to include 34% of the population, by far the largest by number. However, their presence is not just limited to New Zealand since this is a globally established issue.

Survey

The detriments to health issues which nutrition misinformation creates were touched on in the Background section of this report, but to gain a better understanding I sought to conduct a survey amongst my wider stakeholders. These findings will be representative of a social context within my report, thus mitigating them through my solution will contribute to partly the metric by which its fitness for purpose will be evaluated.

Methodology

Danijela's wider clients were selected as the primary survey audience as they encompass a range of people suffering from a range of food related problems within a diverse age bracket (16 - 72 years). As an important ethical consideration, I first asked permission from Danijela to research her clients and she agreed, provided it was done anonymously and her clients had the opportunity to opt in/out.

I also circulated the survey through several discussions I held at my own school's food technology classes. This group was chosen since, given the nature of their subject, I can expect some degree of nutritional knowledge to be present. However, to ensure these two subgroups of my wider food affected stakeholders do not distort my findings of each other I intentionally kept their survey results separate.

Findings

The results of the findings were highly insightful. Of the 63 people I surveyed in Danijelas group 78% said they had attempted to gain an understanding of their food from reading nutritional summaries. The key areas people identified improvements could be made in this context were: '*readability of text*','*complexity of information*', '*inconsistent visual aesthetic across products*' and '*not being able to rely on nutritional summaries for unique foods from cafes and restaurants*'. 94% of respondents said they took their smartphones with them on a daily basis.

Supermarket Contexts

Two key factors that Daniella's group said hindered their understanding of food in supermarket contexts were the complexity of information presented in these summaries and their inconsistent 'unscannable' visual aesthetic. The people who typically read these summaries wanted to conform to either a diet or avoid allergies they were susceptible to. While 36% of people acknowledged they did not bother reading summaries due to their inconvenience most did not say it prevented them from buying the product. 33% agreed flexible packaging was particularly hard to read as information was often lost in crevices and folds. 16% they had asked supermarket employees about nutritional information about their products but found this information was unreliable. The last 15% refused to comment or take part in the survey.

Organic Contexts

I defined the organic context to include places extrapolated from my discussion with Danijela such as cafes, boutique shops, food stores, and restaurants. Both Daniella's clients and the food technology classes felt strongly about the level of nutrition misinformation present in these contexts. Those conforming to a diet (53% - Danijela's Client Group) said they would often have to ask an employee for a nutritional summary. This was often done verbally and these stakeholders felt they could not remember most information when making comparisons between meals. Stakeholders commented that improvements in this context could be made by having nutritional summaries enforced and clearly visible to customers (74%). However, providing a quantifiable means to represent the *'healthiness'* of meals other than just percentages or numbers (88%) was also suggested as an improvement. Although instances where summaries were present some stakeholders felt the information was outdated and not being maintained (62%).

(Note: the survey questions for supermarket contexts were based off a collective percentage while organic contexts were based of the percentage of stakeholders who answered yes to that particular question.)

Reflection and Extrapolation

94% of surveyed stakeholders uses a smartphone. This suggests people are willing to adopt new technological methods and incorporate them into their lives. Although not inclusive of the wider global stakeholder group as a whole, research showed global trends agree with my result and consistently point towards a mobile oriented computing market; *Statista.org*[11] predicts 5.4 billion people to own a smartphone by 2020. Extrapolating, a mobile-oriented solution would have the greatest chance of reaching the widest number of people and meeting these global trends.

Going further however, minimising user mobility in an app based solution designed to mitigate food-related health problems could counter intuitively hinder stakeholder health, especially those suffering from obesity and need to walk more as an app would present information instantly, independent of location. When discussing this with Danijela she said client fitness is often *'below average'*, so encouraging moving, although outside her scope of profession, is still highly important. For me, while designing to take advantage of a mobile orientated solution and the everyday practicality it offers I need to make sure not to indirectly introduce a counter-intuitive negative spiral in discouraging being active.

Outdated meal summaries in cafe and restaurant contexts highlight another relevant stakeholder observation. Outdated summary information misinforms people to what they are eating and highly contributional to my issue, and for that reason does not invoke emotional resonance. In a solution, adaptability to keep up with cafe/restaurant product variability would become important, continuing the importance placed on autonomy. In an abstract sense autonomy also has the potential to lower the time taken for employees to write summaries and would remove the potential for human error entirely.

However, conveying nutrition information without the use of convoluted summaries such as those found in the supermarket was also identified to be important by stakeholders, and especially relevant in pre-packaged contexts. As a future consideration, one possible approach could make a comparison between the meal in question and a relatable food, ie, "*eating x amount of this is like eating y amount of this*". This could relate people to situations they were familiar with to their unknown meal in question and go some way to better-facilitating nutrition education. As a future consideration I could potentially discuss ways to represent meal 'healthiness' with Danijela

Wider considerations

The social-cultural considerations embedded in this problem extend to the sensitivity of food within many of the world's cultures. Stakeholders in this group mentioned the religious significance of pork of Buddhists in India and the sensitivity towards meat for vegetarians, this was verbal feedback and not part of my survey. Regardless it presents both important cultural and moral challenges which need to be respected. Cultural global diversity makes this a particularly challenging consideration to uphold given the extensive nature of the world's cultures.

[11] "Statista." https://www.statista.com/. Accessed 9 Oct. 2018.

Data Scientists

To help me gain a better understanding of future technological challenges I may encounter I elected to source advice from a professional data scientist within the commercial software industry. Through a community programming Meetup I got talking to *Jeff Mo* who is data scientist from Xero. Mr Mo had extensive experience within his field of Data Science and agreed as to his role being one of my key technical stakeholders, giving me valuable feedback as to any concepts, problems or questions I would present to him in the future.

Critical Reflection on stakeholder analysis

In terms of my wider stakeholder analysis I believe my initial research was a success in that it provided me with a wide encompassing view of the may needs that my stakeholders have. The increased prevalence of health issues in people struggling for food related health problems stakeholder group, compared to the average person, suggests the status quo in assuming the needs of the misinformed by food are the same to healthy people is flawed.

From this inference I was able to build an initial understanding of techniques used to analyse food and the requirements relating to time constraints of my stakeholders. Understanding my primary stakeholders knowledge limitations can help me target these issues within my eventual solution. Furthermore, the way I was able to expand my stakeholder group to gain technical information from both a professional nutritionist and data scientist would increase my practical understanding in methods I could employ to tackle such an issue, as well as providing feedback further down the track on my concepts, and development of my solution.

I believe that the high participation rate of Daniella's clients in my survey indicates I can consider my findings from that group of people reflective of people in a similar situation and economic position as a whole. However, it is not an accurate representation of the wider stakeholders in a global sense so I will still need to analyse wider world trends to ensure I am designing to attain a solution with the highest possible degree of fitness for purpose.

Conceptualisation

Diet Automation Idea

Problem Identification

Danijela's Diet Templates
Fig 8 | Source: Supplied by Danijela during research interview

	Breakfast	Lunch	Dinner
Sunday	Hot Cereal; Fresh Fruit	Roasted Veggie & Hummus Wraps; 100% Fruit Popsicles	Carrot Cashew Spread on Woven Wheats; Lentil Chili; Green Salad
Monday	Apple-Cinnamon Oat Squares; Fresh Fruit	Lentil Chili; Salad with Peanut Orange Dressing	Black Beans & Rice Extravaganza; Green Salad; Fresh Fruit
Tuesday	Green Smoothie; English Muffin with Nut Butter	Green Pea Guacamole Wrap; Fresh Fruit	Creamy Curried Cauliflower Soup; Roasted Veggie Couscous; Green Salad

Dieting was the primary technique Danijela used to help her clients. (Fig-8) is the typical format used to communicate her diet programs. It allows the client to clearly see the type of food they should be eating and the time it should be eaten. In situ this table was embedded in a Google doc which was shared with the client and allowed accessibility to a certain extent, however, this was still limited as documents were view only.
I think this also contributed to one-directional nature of feedback from Danijela to her clients. Daniela's prescribed templates were often rigid and based on one instance of sourced clientele. She pointed out that frequent tweaks were made through planned *'checkins'*. I quickly saw this created a detriment to time.

An important method to approximate calorie intake to base these diets off was the Harris Benedict Equation[12] (Fig-9). From this Danijela could estimate a person's **base metabolic rate** (BMR) and subsequent daily calorie intake. Danijela liked this method as it could be quickly executed with minimal client information and generally provides an accurate estimation of a person's energy needs.

Fig: 9	Source: Danijela	Harris Benedict Equation

$$bmr = 66.5 + (13.75w) + (5.003 \times h) - (6.755t)$$

Positives

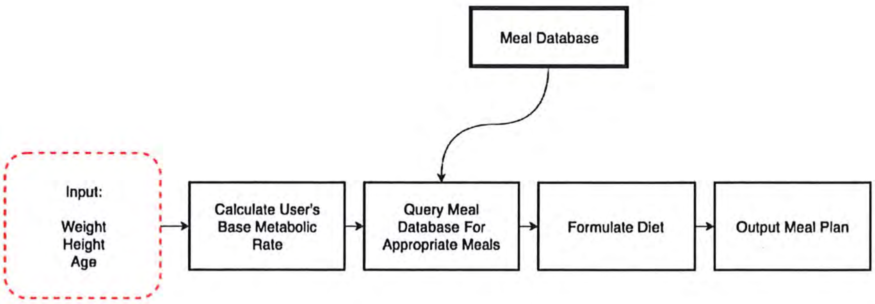
- In situ this table was embedded in a Google doc which allowed accessibility (to a certain extent)
- Clearly shows what food needs to be collected
- Clearly shows when the food needs to be eaten

Negatives

- Making updates to a diet requires a client to come in so Danijela can reassess their weight and health information to base diets thereof.
- Managing and making changes to many google docs for many clients was very time consuming for Danijela.
- Keeping a record of diet changes took a lot of time.
- Diet documents were view only and could only facilitate one directional feedback between Danijela to her clients.

Extrapolation and Synthesis

My initial thoughts on how to best implement my solution were to synthesize and extrapolate from this already proven method of dieting. However, I saw I could improve it by adhering my project's main values in



[12] "A Biometric Study of Human Basal Metabolism - NCBI - NIH." https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1091498/. Accessed 27 Feb. 2018.

autonomy to further facilitate efficiency and scalability. Under this, I would generate a dietary structure which could adapt off a client's changeable health parameters automatically as they progressed through their program and ideally lost weight. My initial conceptual outline of this concept is illustrated above.

Whilst theoretically feasible I grew concerned about how my solution would interpret health data to compute a suitable diet structure based off a client's health parameters. To evaluate this I brainstormed several possible approaches. Figure-10 shows my first attempt in diet generation through a conceptual database lookup system (DLS). Using the Harris Benedict Equation to determine a daily calorie intake could allow background mechanisms to perform a database lookup and find a list of foods where the calorie sum of each meal roughly matched the required calorie count.

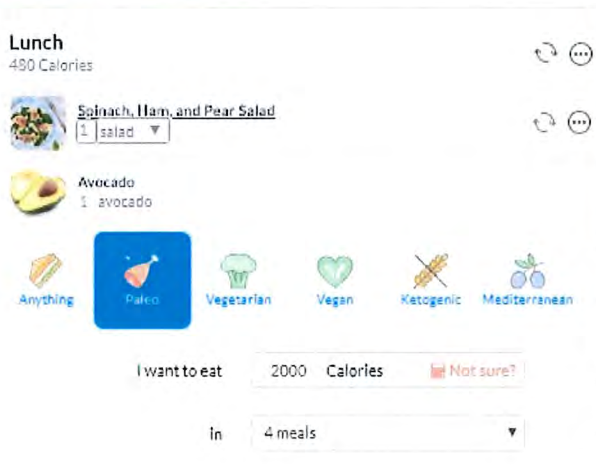
This concept facilitates a desirable element usability as minimal input is needed, an important attribute first extrapolated from my initial client discussion. I presented my dieting idea to Danijela and wider stakeholders. At first, they were impressed with the concept. However, a key view Danijela pointed out was the potential for autonomy to interfere too much with the targeted diet food group. She said rigidity was important to make sure clients stuck to their program, suggesting that although desirable, too much autonomy had the potential to deviate the client from the original goals of the diet. I believed I could manage this complexity by weighting different nutritional properties such as fat and calories in a diet more 'heavily' such that highlighting their importance as the ones that must 'adapt' more slowly.

Another interesting point of feedback was the way I would represent a diet in a readable format. Since most of my stakeholders were used to a fixed diet that could be expressed either through a table or chart I also began to think about how I would show past iterations of the diet with respect to time. Danijela said this medium of data would be important when a client came in for a review as to judge the effectiveness of their diet program. This made me think that perhaps comparing the rate a person was losing weight to the ratio of food groups they were consuming at the time could reveal additional insightful observations and facilitate another medium of education.

Taking these thoughts into consideration provides an opportunity for an improved dieting practice which would:

- Automatic diet generation and refinement would eliminate superfluous meetings between client and nutritionist, saving on time and money
- Impose a flexible and adaptable dieting program which can adapt to a client's changeable health needs.
- Be able to control the rigidity of the program by weighing more important aspects of a diet more heavily.
- Implement an innovative method to show the effectiveness of a diet at any given time with respect to changing diet parameters and the rate at which weight is lost.

Existing Products



"Eat this much" automatic diet generator[13]

Overview

The *eat this much* diet generator is an online dieting system which allows the user to specify how many calories they want to consume. The client can also enter their body weight throughout the program to make changes to their diet. When starting out the user can pick the type of their dieting structure they want to start with and then the system picks out similar meals from the database.

[13] "Eat This Much." https://www.eatthismuch.com/. Accessed 7 Oct. 2018.

Their system also had a estimation system which give the client information about the carbs, protein, and cost they can expect in the real world.

"Eat this much" Diet Generator Reflection

The *eat this much* diet generator already covered everything I had set out to achieve. I, therefore, decided to end development of this concept and choose an idea which would be more unique, looking outside the area of dieting as there are clearly already a lot of solutions, thus minimizing my potential to innovate.

Independent Concept Reflection

My concept also lacks sufficient scalability and the stakeholders I showed it to also felt this way. Danijela raised the point that dieting often includes a specific list of foods that are derived from health benefit it would create, rather than its availability in the client's area. Since people's accessibility to food is often constrained by cost and location this is a flaw inherited from the nature of dieting.

Further evaluation with Danijela revealed the importance of selecting the correct diet to address the client's health issue, stating some had potentially unseen health effects. She said often times it is safer to make lots of small subtle changes to an existing eating pattern rather than one big overhaul. I saw that my conceptual solution would not provide a safe means to mitigate these health concerns as it made no attempt to educate before the program began. It also did not eliminate one of my outlined initial complexities in the economic determinants from seeking a medical professional's help to get this information.

The preceding diet also used simple English words to describe food objects and assumed an understanding of metric gram measurements. Both linguistically and numerically this raises an important social consideration and would conflict with my established wider stakeholders lingual needs.

Based on these factors I believe this concept divulges too far from the fundamental nature of dieting to have a beneficial effect on my explored issue compared to traditional methods.

Automatic Meal Analysis Idea

Background

Historical Trends in Consumer Behaviour

1900s: Large-scale food production within places like supermarkets did not exist 100 years ago. During this period just 10 percent[8] of the world population inhabited cities meaning most people bought local produce from specialist shops. Limitations in transport and infrastructure ment many people produced and ate local food. Furthermore the frequency of low cost purchases was high due to minimal food preservation techniques.

1950s: Agriculture industrialisation increases the scale at which food can be produced. The scope of available food also increases with refrigeration and preservation technologies. These factors in turn attract large supermarkets, thereby through mass production the cost of eating significantly reduces. The low-fat craze that took hold in the 1980s turned out to have unintended and very unhealthy consequences. Food manufacturers cut fat from their products but replaced it with refined carbohydrates, such as white flour and sugar.

Present: Today, over 74% .of the world population now lives in cities[14]. Increased Urbanization means the density of populations rises in turn affecting how we distribute food. This has led to large scale supermarket chains and hundreds of thousands of individually owned cafes and restaurants scattered throughout society. Technologys unremitting evolution continues to shape the way we eat. The advent of Genetically Modified Organisms or GMOs in 2000 now allows us to alter the nutritional content of food to suit specific needs. Chemicals and pesticides are introduced to control agriculture.

---

[14] "Food consumption trends and drivers - NCBI - NIH." https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2935122/. Accessed 10 Aug. 2018.

Historical Trend Reflection

It's clear the way we produce food to support ourselves as a species is contrastly evolving. However, the way new ideas affects consumption highlights to some of the dangers of this evolutionary process. The low fat diet in the 1950s ment many people were detrimentally affected by a widespread misinformation that fat is dangerous and sugar consumption should therefore be adopted. Public misunderstanding led to significant health detriments. Likewise increased of food outlet saturation within society combined with rapid onset urbanisation suggests the customer employee dynamic has also changed.

In the past employee customer numbers were relatively low facilitating a easy environment to ask questions. Today large scale supermarket invite hundreds of people to shop simultaneously while employees are kept minimal to save on operational costs. This reduces a potential of communication within the customer feedback loop and points towards encouraging uninformed eating patterns.

Problem Identification (*Refer to Figs 2, 3 and 4 + above historical trend research*)

Positives

- Health star rankings are a quick way to show the 'healthiness' of a meal.

Negatives

- More in depth meal information quickly becomes convoluted due to;
  - Inherent complexity. Numeric and linguistic lingo makes it hard to understand/relate to customer context.
  - Presentation: text size, curved or flexible surfaces make summaries hard to read.
- It is hard to abstract summary information to other areas.
- Inability for static summaries to adapt to changeable and unique contexts such as food from cafes, restaurants or home cooked meals means they are expensive for food manufactures to reproduce .
- It is in the economic interest of some corporations to hide health information on unhealthy products, making it hard for consumers to understand or find.
- Food diversity means summaries are often printed at different locations on packaging making it hard to find a consistent place to look.

Taking these thoughts into consideration provides an opportunity for an improved food consumption practice which would:

- Facilitate a scalable technique to provide health information about any meal in the world.
- Present information in a way that is
  - Displayed in the style to maintain a degree of continuity in terms of aesthetic and understanding across different contexts.
  - Easy to interpret.
- Educate and inform why customer foods are beneficial or detrimental to their health.
- Allow for global deployment through multi-numeric metrics in feedback.

Extrapolation and Synthesis

Looking at my previous diet idea objectively, the problem did not so much lie in the method people would take to lose weight, but rather the way it encouraged people to abandon their existing eating patterns without explaining why.

I saw that I would need a solution which was informative, pertaining to my wider projects value of education and scalability, so it could be applied to any person or meal. This new idea came to me based on my past experience as a Xero[15] intern since I had become fascinated by Machine Learning. ML vision systems can closely mimic and emulate the observational vision capability of humans, thus, harnessing this shift in the way we use technology can be an excellent method to solve real-world problems. Computer vision particularly invoked emotional resonance. Enabling a computer to 'see' its environment made me

[15] "Beautiful Business & Accounting Software ...." https://www.xero.com/us/. Accessed 25 Oct. 2018.

think how we could teach machines which foods are beneficial to someone's health, hence educate them about what choices to make so they, in turn, could teach their human users.

With this in mind, I thought back to my research interview with Daniella, specifically her technique in *object based analysis.* I conceptualised a solution which could recognise the type of food a user wanted to eat based of a provided photo and then perform an autonomous nutritional analysis to output a self calculated nutritional summary detailing a calorie count and energy contents. Using this information then a recommendation explaining why this food was beneficial or detrimental to a clients health could be automatically made.

Stakeholder Feedback

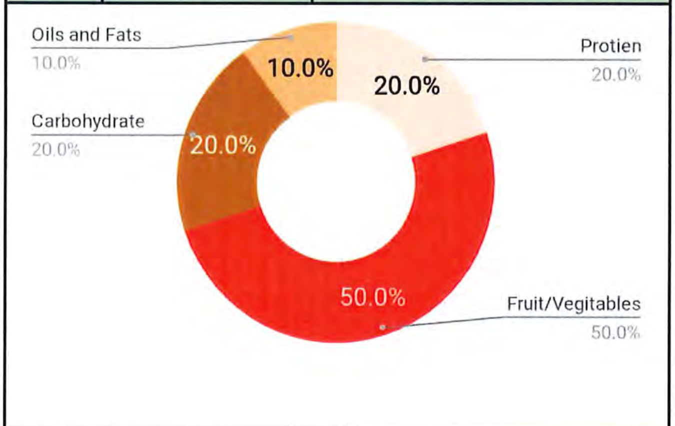
Consultation with Danijela

I discussed my proposal with my key stakeholder. The main points of this discussion are listed below.

Method of Education

An interesting point Danijela raised was the way my end product would visually communicate the 'healthiness' of food. I had initially believed it best to represent within some sort of percentage or scoring system. However, Danijela mentioned from her past experience having an arbitrary percentage or even binary *yes* or *no* as an indication of healthiness can be disheartening and detrimental to a person as it did provide any explanation *why* the food was unhealthy. I agreed and also saw it undesirably mirrored a typical nutritional summary.

Fig 13 | Source: Danijela | ***Danijela's Meal Portion Ratio***



Danijela stressed the importance she placed on educating her clients about healthy food choices so that they can later apply those new skills independently in the future. As an improvement, a more suitable approach suggested was how she herself analysed her clients meal structures: by comparing them to an idea case represented by her portion ratio (fig 13). Through this method protein, carbohydrates and vegetables were weighted different importance. Danijela explained that these values for this ratio acted as a base indicator and would often be tailored to her client.

Accuracy

We both established an automated system needs to be accurate to the highest possible degree to make sure it does not counter intuitively misinform people about what they are eating. Presenting inaccurate or outdated information was also a negative detriment pointed out in my food affected stakeholder group survey.

Technological Medium of Solution

An important point Danijela raised was ensuring this idea could be mounted within an app. Believing this was appropriate and would suit the growing technological trends within her area of the industry. Danijela pointed out that a digital solution had the potential to introduce an element of convenience amongst her clients. (*This was assuming my concept would eventually be commercially implemented.*) I understood this particular perspective thinking back to her google doc method in communicating diet templates.

Drastic Changes to Eating Patterns

Danijela also suggested that I discourage users from making radical diet changes She said that completely switching up a diet had the potential to invoke potentially negative effects on client health. Instead I should encourage people to make lots of gradual small changes through an informed and educated approach to nutrition.

Discussion with Jeff Mo

To gain a better understanding of how to approach designing such a solution I consulted my wider technical stakeholder, Jeff Mo.

Use of machine learning

Since I had previously discussed my idea with Danijela I was able to refine what I actually wanted to achieve in this solution. While talking to Jeff we went further and identified the complexity of my addressed issue in both the infinite diversity of people's health and types of food.

I explained to Jeff I wanted to create a system which could analyse people's meals by providing them with a list of ingredients, and, ultimately, the end goal of making a recommendation as to whether that food would be healthy. Although incorporating AI was at the back of my mind I initially suggested to approach the problem by means of some sort of food or meal API that could give a generic list of nutritional properties for generic types of food. One I had come across in my research was *Spoonaculars Food API.* It contained 360,000 recipes. Under this approach a user could enter the address of their restaurant and Spoonaculars Food API would automatically lookup that restaurants website to try and find their menu.

Despite the impressive scope Jeff pointed out conventional approaches to solving this sort of problem would be too rigid for my needs. I saw the importance of this due to the aforementioned understanding of extensive stakeholder context and diversity. As a more adaptable approach, to the future and ourselves, he suggested I look into artificial intelligence and specifically: machine learning.

Why Machine Learning?

We had already established I would need a program that could draw a distinction between *healthiness*, *ingredients* and a *picture of food.* But the variability between these factors is simply too complex for a human or static API to account for in all possible permutations of this problem and encapsulate it in an effective solution.

The use of a *neural network* leverages the advantages to adaptability as part of a shift in technological problem solving first touched on in the introduction section of this report. A Machine analysis solution would also remove the possibility for human error and function independent of external input, minus the need of a photo of food. These attributes remain inline with efficiency and ease of use extrapolated from my discussion with Danijela and continue to be important values identified thus far.

Training Dataset and Granularity

Jeff pointed out the most important process to delineate an AI through machine learning was training. He stated you basically show your training algorithm lots of labelled examples, in situ this will be food images, so over time it is able to infer patterns, synthesise inferences and extrapolate this knowledge to new unseen contexts. (I later learned this process is called *supervised learning*)

Taking this into consideration for training my solution, as well as just images, I also saw the need to include recipe data to give the AI a better idea of nutrition analysis. This will also be important in order to make a comparison against Daniela's portion ratio possible.

This made me think to successfully facilitate training at the scale I am intending *granular classification* would become very important. I raised this with Jeff and he explained this will be dependent on the detail I can achieve in my training dataset. Since the output of the neural network will directly mirror that of my dataset the information needs to best represent my stakeholders' context. Therefore, as future consideration, understanding what my stakeholder context will visually look like, as well as how it affects them and machine made meal inferences, each become fundamentally important.

Computational Envelope

A neural network would be required to run efficiently within my physical environment and accommodate the entire spectrum of high end and low end mobile devices to remain fit for purpose and inclusive of stakeholder needs in a physical and social sence. Yet research up to this point, as well as my own personal experiences, proved running neural networks is very computationally expensive. Due to this I had originally thought of running the neural network in a remote server and then passing interpretation data via the internet back to the user was the answer to this complexity..

While Jeff acknowledged training a neural network is intensive he pointed out training is often carried out off platform within a specialised environment. Once a network has been trained it can be optimised for my physical environment as a smartphone, making it significantly less intensive to run.

I saw this modification could facilitate both offline performance and minimise analysis execution time. Offline functionality would eliminate the need for an internet connection and thus another economic detriment to my stakeholders. Minimal execution time would also increase solution functionality and fitness for purpose within time pressured contexts such as the drive thrus and fast food restaurants I had talked about with Danijela.

Synthesis of Stakeholder Feedback into refined brief

Customers should be able to determine the nutritional benefit of their meal through a single photo:

This could be achieved by:

- Incorporating a app and user interface which will display nutrition information about a meal and an overall judgement of the meal's healthiness.
- Incorporating a neural network to make meal predictions and infer health data

Must educate customers in a way which mirrors Daniela's meal portion ratio comparison technique:

This could be achieved by:

- Making a comparison between an analysis case and the portion ratio to show a user areas of their meal that are lacking beneficial nutritional properties.

Ensure customer safety when using the product:

This could be achieved by

- Encourage safe eating patterns by minimising drastic drastic changes and encouraging lots of small minor changes
- Making sure analysis output is accurate so customers are correctly informed about the food they are eating

Facilitate Global Scalability:

This could be achieved by

- Ensure a highly detailed training dataset, reflective of the worlds foods, so to be able to provide nutritional and health feedback on any meal in the world.

Dataset must reflect real world stakeholder contexts

This could be achieved by

- Structured around *ingredient* to *food image* example pairs for training the neural network.
- Ensure a highly granular/detailed dataset structure for the purposes of ensuring high accuracy.

Consider the computational envelope of a user's device

This could be achieved by

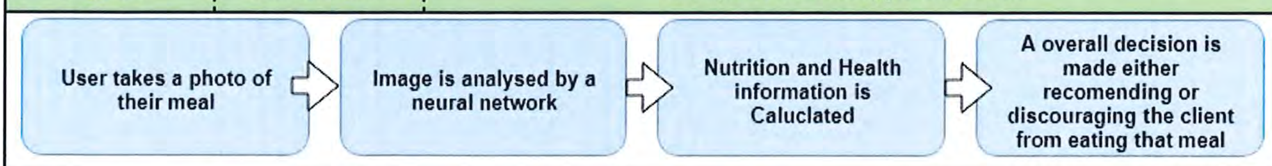
- Taking into consideration how camera resolution could affect machine analysis in terms of accuracy
- Consider my solutions impact on battery life, ie, minimising on energy draining tasks

Proposal

Outline

I propose to develop a solution which automatically interprets food information, no matter the context, type of food or user location, based off a single image. The system will be created so customers can download an app to their smartphone, take a photo of their food and have it automatically analysed to provide a summary of nutritional content and also make a recommendation to the user whether or not this meal would be beneficial to their health. My conceptualised use case is shown in Fig 14.

Fig: 14 | Source: self | **Proposed Prototype Use Case**



Full Project Specifications

Issue Summary

My identified issue is the rising problem of food related health complications as a result of people going uninformed from their eating habits.

Proposed Outcome

User interface

- Be intuitive to use and friendly to maximise stakeholder participation
- Provide the customer with sufficient information to enable them to understand the sorts of food they are eating while still using conventional everyday vocabulary in explanations.
- Use Danijela's meal portion ratio

Meal Classification and Analysis

- Must function independently from a internet connection.
- Analysis must be accurate.
- Comprehensive to help people make informed decisions about their food whilst still being interpretable by the average (wider stakeholder group) person.
- Must uphold my project's values derived from prior research and stakeholder feedback in:
  - Autonomy - to ensure solution scalability and adaptability in a diverse range of potential analysis cases
  - User Education - must inform users *why* their food is beneficial/detrimental food related health complications
  - Efficiency - analysis duration must be kept at a minimum

Physical Environment

- Portable, be able to run on a range of computationally variable mobile devices
- Accuracy in meal analysis cases should be independent of camera resolution.

Social Environment

- Appeal to the economically disadvantaged by:
  - Being able to to run on both low end and high end devices.
  - Be able to function in an offline environment
- Consider the social-cultural sensitivities associated with food

Development

Food-Ingredient Dataset

Component Outline

The food-ingredient pair dataset is an important component of my greater meal analysis system because this collected information will be used to train the neural network on the fundamentals of image classification and nutritional interpretation. For this reason, although it will not be physically present in the end product, it is still fundamentally important to develop and consider.

Continuing with my research and discussion with Jeff Mo showed training data comprises of images each tagged with what they contain. In my instance this tagging will include the meal name and respective ingredients. This is a crucial step for the algorithm to infer instance specific features, thus any poorly tagged images gathered during the dataset creation process could negatively impact on the neural network's accuracy and would not suit my stakeholders needs in informing their eating trends.

I had already established the importance of a highly granular dataset to facilitate global scalability. However, from past experience building such a large dataset which strived to encompass a vast range of food categories whilst still maintaining the necessary depth to achieve accurate classification would quickly encounter problems in disk usage in the storage of these images. It was clear the necessary storage capacity be established before data collection would begin.

To facilitate my brief specification of accuracy I also needed to think about the type of food images I would collect. Whilst images taken under idealistic laboratory conditions would obviously be the easiest to work with real world images of food taken in context are far more likely to represent my stakeholders context in situ and the wider physical environment, yielding more meaningful results during training.

Preliminary Dataset Specifications

- To be as similar as possible to the actual images which will be encountered in the real world.
- Facilitate granularity in the breadth of total food groups accommodated in the dataset to the depth of images for each group.
- Each image group should contain a variety of photographic and environmental conditions relating to specified food group

Data-set Collection Methodology

My chosen mechanism to data collection will need to pertain to my outlined specifications. The two I have identified are *using an existing dataset* or taking a *web scraping/data mining* approach.

Using an Existing Dataset

After discussing my project idea with Jeff Mo, he suggested I start by looking at existing datasets, highlighting the *Food 101* dataset which he said is often used by researchers when dealing with food related research problems. Food-101 contains images sourced from real world contexts, better reflecting my stakeholder's intended physical environment, making it stand out since after extensive searching of other datasets it was frequent to see images taken under idealistic laboratory conditions which to not represent my intended stakeholder physical context.

However Food 101 only contained 213 different types of foods and thus could only guarantee minimum granularity. Justifyingly this sort of dataset would not be able to facilitate a globally scalable solution since 213 classes of food can not accurately encompass the extensive demographic tastes of my wider, globally based, stakeholders.

Moreover, to develop a comprehensive mode of food analysis my neural network should be able to derive ingredients, calories and nutritional information. This is an integral part of my solution so I began to worry about the absence of this ingredient data in Food-101, or any of the other data sets I evaluated. Initially I believed I could map ingredients from an external source such as a recipe database, to each meal class across the whole dataset. I thought I could automate this process in Python. But as research continued I began to doubt the practicality of this technique in both the time taken to conduct ingredient mapping and the resultant accuracy for which ingredients pertained to what foods.

Methodology Evaluation

Benefits

- Minimises the time needed to collect training data.
- Clear representation of the food classes in question.

Detriments

- Lacks ingredient-image training pairs.
- Can not conclusively facilitate dataset granularity due to minimal breadth of food types

These factors contribute significantly to my decision that using a pre-prepared dataset would eventually come to negatively impact my solution in terms of its fitness for purpose in the broadest sense. Given the complexity of my targeted problem I had expected this to be the case.

Web Scraper

I went back to Jeff to discuss other possible approaches to data collection. Jeff pointed out the technique of web scraping which is the automated harvesting of data from web sites.

An advantage of building my own web scraper would be that I could taylor it specifically to my needs. I could target a specific website and tell the scraper it to download images from pages that contained meals.

A balance I will need to find during scraping is the number of different foods (breadth) to the number of images for each particular type of food (depth). This ratio will determine the number of different recognisable food groups and accuracy of each group. I can expect this accuracy to be proportional to that of the number of food groups I can encompass in my dataset based around findings demonstrated within Alex Krizhevsky's ImageNet Paper[16].

Methodology Evaluation

Benefits

- Customisable pipeline means I can tailor it to my intended needs
- I can target specific websites which have data that represents my wider stakeholder context
- By scraping recipe websites I can efficiently collect images of food which have an accurate list of associated ingredients.
- Collecting data from websites ensures images are sourced from all over the world, better representing my wider stakeholders demographic and beneficial to a scalable solution

Detriments

- Would require a lot of time to develop and a lot of physical storage space for collected images

I briefly considered merging a pre-built dataset along with my own images collected from a web scraper, taking the best of both worlds and potentially adding a desired element of photographic diversity. However, this approach was quickly dismissed as I realised merging two completely different file structures, along with management of mismatching food groups, file type, and resolution would be extensive and protrude too far beyond the scope of this project. By these terms I felt I was able to justify my decision to solely use a web scraper as my method and material taken to collect training data.

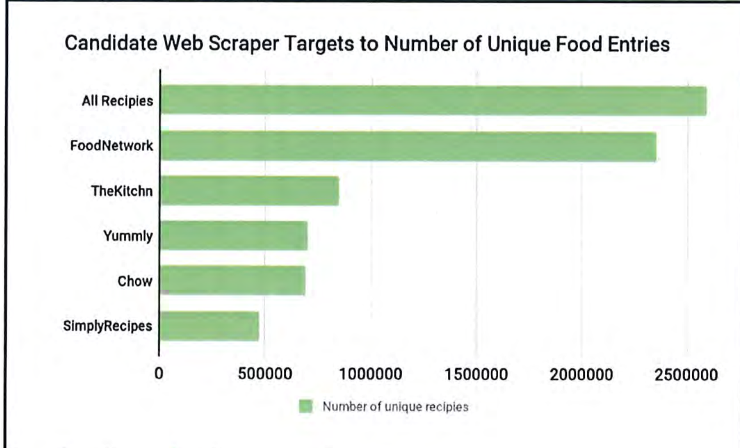
---

16 "ImageNet Classification with Deep ...." https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks. Accessed 5 May. 2018.

Web Scraper Development

Choosing a Scrape Source

Fig 15	Source: Self	The proportion of meals per recipe website



Data: http://www.ebizmba.com/articles/recipe-websites Source:Self

Given image-ingredient example pairs were an important specification necessary for training, extrapolated from my discussion with Jeff, pairing ingredients to images to produce trainable examples was particularly challenging. I had initially considered targeting Pinterest and Google Images because they held a wide diversity of food images. However, they lacked the relevant ingredient data.

After much research, it tuned out the answer was simple and could be found in Cooking and recipe websites.

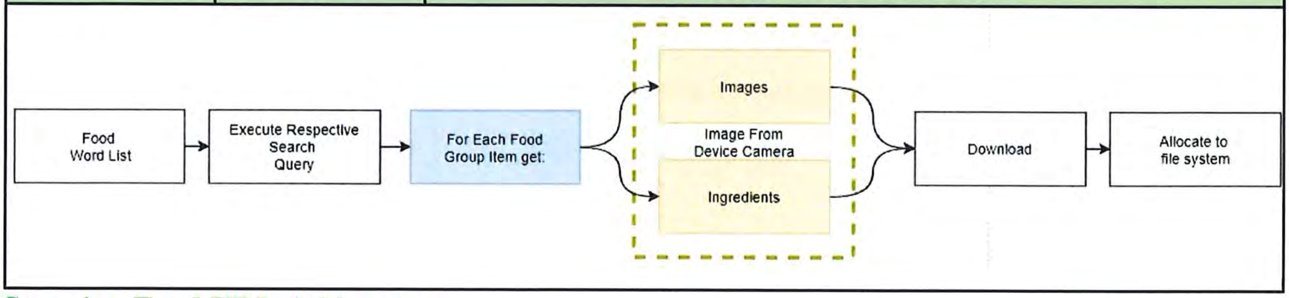
These websites were ideal as they contained a large quantity of relevant images of food with respective ingredients on the same page. Ideally, once an image and its ingredients have been identified, scraped from a website and correctly formatted they can be stored on my local file system within a related food group, represented by a directory.

Choosing the website where I would scrape image-ingredient data from was another complexity I encountered as this decision would directly affect my neural network's performance. After researching extensively I chose to target *Allrecipes.com*. Firstly is a recipe website. It also had the largest scope of unique food dishes of the sources I evaluated (Fig-15). For this reason I believe it has the best chance of fulfilling the emphasis I have placed on image granularity. The demographics of the source also make it ideal in addressing global scalability as its global community of users is reflected in the 23 different languages which the site supports. I could therefore assume Allrecipes.com would render the largest, most diverse source of data to collect.

Approach to Scraping Data from AllRecipes.com

I then began research how best to implement web scraping functionality into a deployable tool to aid in my collection of training data. I already had extensive knowledge in working with C# to develop Windows based desktop programs, however the process of web scraping was very new to me. To visually represent the tasks I needed this tool to do I created the below flow diagram.

Fig 16	Source: Self	*Web Scraper Functionality*



Scraping Food URL Addresses

Since Allrecipes.com, or any of the websites I evaluated, did not provide any sort of API to interface with the content on their websites I needed to find a way to automatically query the website through code. After much thought I realised could take advantage of Allrecipe's tendency to define meal query keywords in its url, rather than handel them on it's server side. I could manipulate this URL to specify which type of meal I wanted to get meal data on. Furthermore I could define the sort order or recipes in terms of popularity (&sort=p), best match (&sort=b) and most recent (&sort=r). The food word list in fig-16 would be used as the content for each query and would be a text file containing a diverse list of foods to establish the preliminary breath within the scraping tree.

URL template:

```
string URL = "https://www.Allrecipes.com/search/results/?wt=" + foodList[i] +
"&sort=p";
```

Situ:

```
https://www.Allrecipes.com/search/results/?wt=cupcake%20%20Peanut%20Butter%20an
d%20Jelly%20Cupcakes%20&sort=p
```

I decided this was the best method to get data from allrecipies.com as it allowed me to define what I was searching for and specify the relevancy of results. I could also correlate collected results against each query's keywords to determine which collected images pertain to which food group. Below I show several recipe urls collected after querying "*peanut butter and jelly sandwiches*" .

Fig 17	Source: Self	*Collected Meal URLs based on a specific query*

```
https://www.allrecipes.com/recipe/49943/grilled-peanut-butter-and-jelly-sandwich/
https://www.allrecipes.com/recipe/23097/ignacios-super-peanut-butter-and-jelly-sandwich/
https://www.allrecipes.com/recipe/26244/uncle-macs-peanut-butter-and-jelly-cookies/
https://www.allrecipes.com/recipe/185951/peanut-butter-and-jelly-thumbprint-shortbread-cookies/
https://www.allrecipes.com/recipe/221964/spicy-pbj-wings/
https://www.allrecipes.com/recipe/11379/sugared-black-raspberry-tea-cookies/
```

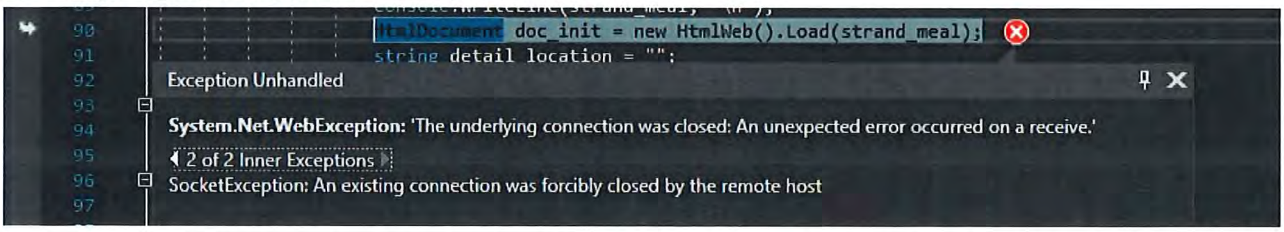
Downloading Meal Data

Now that I could get a list of urls which all contained roughly the same type of meals from a single query (Fig-17) I moved on to developing a way to download this data. When talking to Jeff about this he said it is good practice to structure training data with consistent image dimensions. However, allrecipes contained a range of images of different dimensions so Initially I thought I would have to do download each image first and then resize it using some sort of graphics library. However I eventually realised that allrecpipes.com specified the file dimensions inside each image URLs. By simply setting the width and height dimensions of each image url I could manipulate size before downloading, saving me a lot of time:

```
https://images.media-Allrecipes.com/userphotos/250x250/5305574.jpg
```

detail_urls.Add(sub.Replace("424x655", "512x512"));

During the downloading phase I encountered a ethical consideration I had not considered. Since my scraping program was constantly querying Allrecipes, at an average race of three per second, it was putting a lot of strain on the site. This led to my ip address being temporarily blocked, I suspect by their DDoS protection service, giving the resultant error on my end:.



To manage this I limited myself to download images in periods of low website traffic and at reduced set intervals to minimise the load.

Storing Data

Based off the preliminary test I noted the file size of the data I was collecting. Each downloaded image had the same file type and dimensions so each roughly took up 1mb of disk space. During a short 15 min test I had gathered 933 images, and the total disk space used was 1.98 gigabytes. Although small I thought to when collecting hundreds of thousands of images the physical disk space used would be significant. While not ideal, Jeff Mo and I both feel that the vast majority, if not all, data for my project will be gather this way and therefore significant disk usage is unavoidable. I connected an external 2TB drive to my PC and set the scraper tool to download there.

Preliminary Results

Initially, I was pleased with the first few tests. After several hours I had 2341 images. The results of the initial test are illustrated below in a 14x8 sample of the *'fruit'* food group.

However, I quickly identified minimal granularity as an issue. This was dependent on the extensiveness of my food vocabulary list that queries were based off, which, in hindsight was fairly limited as it only contained 83 food types.

Moreover, there was no delegated specificity within the different types of fruit items. My current approach could not differentiate separate types of fruit into their own sub classes. For example I can see *avocados*, *seeds* and *carrots* have all been grouped together. This grouping destroys any benefit granularity would create in the first place. If I were to take this data and train the neural network on it I would expect to lose the diversity of classification within this fruit group, and inevitably, wider word tests. This detriment would also negatively impact my social environment as a lesser accommodation of analysis diversity would inhibit food groups from being recognised in different cultures.

The collected images also lacked qualitative diversity as a proper representation of my stakeholders context. All images are very clear, well-lit and framed on the food item in question. Reflecting on this I think it's because Allrecipes orders food by popularity which puts sponsored listings and idealistic cases at the top. This is problematic because it minimises important environmental factors such as noise, lighting ,context and reduced clarity. In some respects accounting for this 'noise' in the data is an important physical consideration for my dataset as the algorithm will need to learn to function within variable environment conditions. This also means if I am able to train a classifier to a sufficient level of accuracy, then I could expect its predictions to be more robust when used on real-world data.

Problems with my dataset at this point

- Granularity is minimal
- Minimal subclass separation
- Minimal representation of data to the real world and stakeholder intended context

Adaptations

Solving the complexity of delegated granularity was particularly challenging since I was limited by the specificity of my food vocabulary list to which I was querying Allrecipes. Manually adding more items to this list would not be technically feasible as it would be impossible to come up with a list of every food in the world. Additionally, food globally is constantly evolving, so having a static word list would introduce the detriment of eventual outdated meals. I considered keeping my vocabulary list the same but increasing the number of images I collect for each group. In this way the bot could eventually encounter varying food types within the same query. I tried this but was displeased at the rate I was capturing diversity.

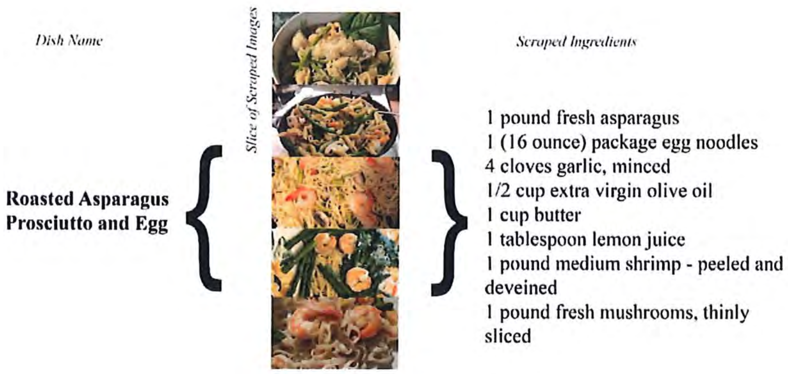
After much thought I came up with a solution that would take advantage of the '*suggested dishes*' section from Allrecipes.com. This provided 9 similar meal suggestions bearing variants to a user imputed query but were still different enough to be considered a different type of food. With this in mind I adapted my approach. I created a integer variable called depth and in the scraping pipeline I would scrape the site's suggested dishes, and the suggestions for each of those, and the suggestions for each of those (continuing on until the current scrape iteration was equal to 'depth'); I would then move onto the next food group in my vocabulary list. By doing so I exponentially increase the number of collected images by a power proportional to my 'depth' variable.

With an initial vocabulary list of 89 food types, with a depth(d) set at 3 I would end up with:

$$89 * (9\hat{}3) = 64881\,\textit{food groups}$$

64881 initial food groups would be a substantial amount to work with, yielding sufficient granularity.

Reflection and Evaluation



The results of putting this process into practice are partially illustrated left. Where the query 'cake' returned a vast array of similar yet slightly different variations of cakes such as '*golden rum cake*', '*carrot-cake*' or '*tiramisu-layer-cake*'. Note, each image represents an individual food group class within the subset 'cake' .

The collection of ingredients was also successful. In this example: '*roasted asparagus prosciutto and egg*' has a comprehensive ingredient summary collected. It's worth noting as a future consideration these ingredients use imperial measurements.

Additionally, by scraping recipes and images from accounts of real people, rather than aforementioned sponsored recipes found on the website, this allowed me to gather more realistic data in everyday contexts. This better situates my project in functioning within the physical environment of my stakeholders as chaotic image conditions are better accounted for, ie the spoon in '*black-magic-cake'* or the tinfoil in '*white-almond-wed'*.

Looking closely there appeared to be irrelevant entries on the Allrecipes site that had been transposed into my collected data. For example the '*dirt-cake-i*' flower pot. Browsing the site before scraping I had expected this to happen. While not ideal both jeff and I agreed that the minimal effect this would have on actually inference performance would not justify the effort of filtering these irrelevancies out.

Formating Dataset

Splitting Data

To effectively work with my collected dataset it was necessary that I slice the dataset into subsets for training, testing and validation. This is an important step as it will mean that separate images to not mix, paramount in preventing the neural network from overfitting (explored below) and reflecting the neural networks true performance during testing.

The ratio of images per training, testing and validation sets is another important consideration. After extensively researching this complexity I based my decision according to Google's outlined data splitting guidelines[17] which were dependent on two parameters. Firstly, the larger I make my training set, the better the NN would be able to *'learn'* in terms of inferring predictions with a greater accuracy. Conversely, the larger I make my testing set the tighter I will be able to define my confidence intervals relative to the prototype's perceived accuracy. Since my dataset contains a finite amount of images these two dependencies are constantly in tension so finding an applicable balance will be critical.

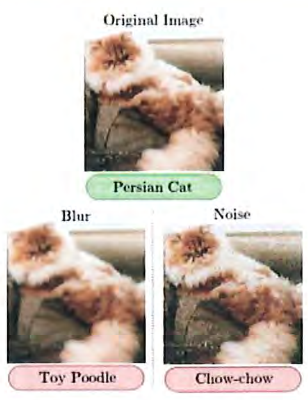
When defining this slice distribution ratio I must consider each set is large enough to yield statistically meaningful results, and that both subsets of data are valid in terms of their accurate representation of the big picture as a whole, *as* my stakeholder's physical environment. By these terms I felt that a 7:1:2 ratio would best fulfill this purpose.

When downloading my scraped images I noticed their order mirrored allrecipies.com in terms of popularity. This made me think about the potential for a bias to arise within training data. If what made these images popular was their high resolution or a particular food group I did not want that pattern to be transposed onto my neural network during training. After researching I saw it best to randomise the order of my images once downloaded. Extrapolating from further consultation with Jeff proved supportive of my decision. Jeff stated that randomisation is simply a must because it gives the network a better estimate of the loss function and its respective gradient. If I was to skip over data randomisation it could lead to a biased representation of the direction of the loss surface to descend down, leading to suboptimal performance.

Complexities relating to dataset variability

Distortionate Factors

Fig 25-b ***Distortion impact on classification performance***



Source: https://arxiv.org/pdf/1604.04004.pdf

At this stage in development, it occurred to me the potential for variable smartphone camera resolution to introduce inconsistency access use cases, a variable if not properly accounted for might distort performance, raising a relevant social consideration. My prototype should function well on both expensive and cheap smartphones, independently of camera resolution.

This anomaly[18] is further explored in Fig 25-b as various distortion cases such as blur and artificial noise reveal the impact on classification accuracy for the CIFAR-12 Neural Network. Although these distortions are far from the worst case they illustrate a specified difficulty neural networks have in recognising the correct class on unexpected input. I can account and manage this by mirroring these disportionate factors in my training data in order to 'teach' the network they do not affect the classification result.

[17] "Training and Test Sets: Splitting Data | Machine Learning Crash ...." 16 Feb. 2018, https://developers.google.com/machine-learning/crash-course/training-and-test-sets/splitting-data. Accessed 17 Mar. 2018.

[18] "Detection based transfer learning." 17 Jan. 2018, https://arxiv.org/pdf/1802.02185. Accessed 15 Mar. 2018.

By applying these post-processing techniques in different combinations in the same image I can account for these variabilities as well as possible. It is not possible to reflect on this consideration as I am yet to build the Neural Network, as thus the benefit this actually creates is discussed in my Initial Evaluation

Fig 25-c | Source: imagu | *Image distortions*

To manage this complexity I implemented *Imaug* which is a Python image manipulation library that randomly manipulates a given image in terms of its resolution, color, transform, rotation and blur.

By applying post-processing techniques in different combinations on the same image Fig-25-c I can account for these variabilities as well as possible. It is not possible to reflect on this consideration any further at this point as I am yet to build the Neural Network, as thus the benefit this actually creates is discussed in my Initial Evaluation.

Overfitting

Another significant complexity inherent with developing my neural network is managing overfitting. Overfitting is when a NN identifies patterns within a training set too well to the point where it negatively inhibits performance when new unseen data is introduced. For example, a instance of overfitting could be if all the images inside a training set were taken included a red table top. A classification model might infer that the table top colour in the image might in some way influence the meal's associated ingredients. Therefore it is clear that just as to manage disportionate factors above a similar extent of randomisation is needed to manage this complexity.

Neural Network Development

The neural network is the most important component of my solution as it is the brain to which ingredients, nutrition recommendations and meal information will be calculated.

Component Outline

Discussion with Jeff Mo

Before beginning development I discussed with Jeff what the best approach to constructing the neural network would be. I had already established input data would consist of a picture of food and the output would contain the meal name, ingredients and associated health detriments. Jeff suggested I first develop a suitable representation for each of these components, pointing out Joint Neural Embedding. Researching this I found joint embedding is composed on an encoder for each modality of training data, in my case this will be the aforementioned meal name, ingredients and associated health detriments. Encoded training data saves on space and is then fed into a conventional training algorithm. I also found learning cross-modal embeddings from additional recipe-image pairs I had collected from Allrecipes.com could be incorporated during the training process to strengthen inferences made and enable the NN to achieve in-depth understanding of food from its ingredients during classification.

Another important point Jeff raised during our discussion was convolutional neural networking (CNN). A CNN is a specific type of neural network architecture composed of a series of layers where each layer defines a specific computation on an image. This approach takes advantage of an input always consisting of images and constrains the architecture in a more sensible way. Unlike a regular Neural Network, the layers of a ConvNet have neurons arranged in 3 dimensions: width, height, depth. Jeff explained in theory this allows for each RGB channel of an image to be analysed simultaneously which in combination with the joint embedding leads me to believe the CNN would become the encoder for image data to meal name.

Discussion Summary

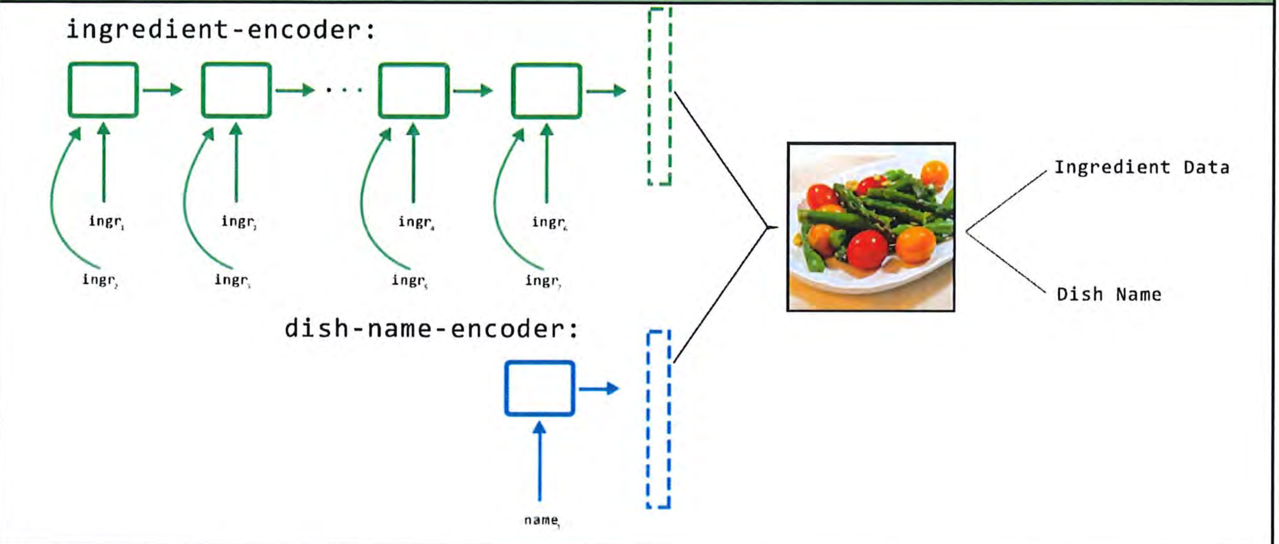
- Using joint model embedding of meal name, ingredients and health detriments would be the best approach to provide the necessary analysis data to my stakeholders.

- Using Convolutional neural network would be best suited as the image-meal name/ingredient encoder as it is designed for visual image inferences.

Network Architecture

To better visualise the conceptual structure of the neural network I created the below diagram based on my discussion with Jeff.

Diagramatic representation of ingredient and dish label encoder as part of the joint embeddings approach.
Fig 26 | Source: self



In it I outlined the low level functions to handle the encoding of recipe data into the neural network model, which was important for the aforementioned joint embedding approach. Fig 26 shows how these functions interact with a meal instance. (Note, Recipe data includes the dish name and its ingredients, (health detriment data is omitted for simplicity)).

I expect (this is tested later) the approach of modularised ingredient components, made possible through joint embedding, to better facilitate my projects value of accuracy. This is important because I want the neural network to be able to identify ingredients in one dish and same ingredient in another dish of the same type but in a photo not encountered during training. This would be far more effective than merely encoding a static label and ingredient description to each class in the model during training, and further functionality within a diversity of stakeholder contexts.

Building the Ingredient-Image Encoder

Synthesising on the pretense of knowledge gained from my discussion with Jeff I knew convolutional neural networking would become a key technique in the ingredient image encoder, as part of my wider training algorithm for the neural network. By treating image input data as a three dimensional volume offers several advantages over conventional techniques. Since a tensor is a generalization of vectors and matrices I found information can be represented within a n-dimensional array of base data types. This facilitates increased optimisation because lower order information such as ints, bytes and strings types can be allocated lower dimensions in the network while higher order objects such as images and vectors can make full use of their respective n-dimensional space.

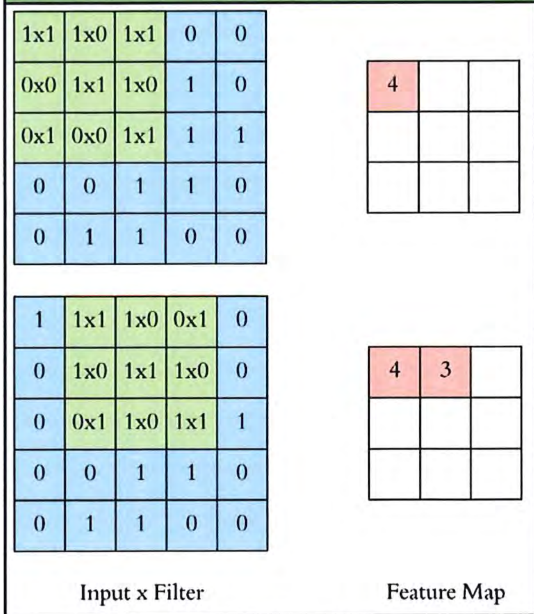
This technique can be seen below where I take a 2D image and convert it into a three dimensional tensor.

```
with tf.name_scope("input") as scope:
    input_img = tf.placeholder(dtype='float', shape=[512, 512, 3], name="input")
```

Conversely, I use a one dimensional tensor to represent the scalar nature of list of integers which represent the string name for each food class in my training data.

```
with tf.name_scope("target") as scope:
    input_labels = tf.placeholder(dtype='string', shape=[number_of_classes],
name="Targets")
```

Conceptual Convolving Operation
Fig 27 | Source: *Jeff Mo*



I then moved onto developing the layers of the ingredient encoder. During convolution as the network convolves each filter across the width and height of the image it also takes into consideration this extra depth dimension. Using this data the dot product between the current receptive field of the filter (light green box in Fig 27) and the input for any spatial position can be calculated. Transposing 3d space to 2d is important because stacking these computed activation maps produces a series of 2-dimensional activation maps that give the responses of that filter at every 3d spatial position. This approach is beneficial since the network will identify filters that activate when they see some type of visual feature such as an edge of some orientation or a blotch of some color on the first layer, or eventually entire foods on higher level layers of the network.

This technique is more effective compared to conventional 2d methods of image analysis because the original image is transformed layer by layer from the original pixel values to form the final prediction scores.

Fig 27-a	Source: Self	*Convolution Layer Implementation*

```
def conv_net_att(self, in_layer, out_size=32, feature_size=(32, 32)):
   input_size = [batch_size, 512,512,3]
   b = tf.Variable(tf.random_normal([out_size]),name='conv')
   w = tf.Variable(tf.random_normal([feature_size[0], feature_size[1], input_size, out_size]),
name='weights')
   conv = tf.nn.conv2d(in_layer, w, strides=[1, 1, 1, 1], padding='SAME')+b
   return conv
```

From past experience I had found it more efficient to analyse images when the file type and image dimensions are consistent as this allows for a pre-determanable learnable filter size. These dimensions are carried through all instances in the ingredient encoder, so if one were to change all need to change. I therefore quickly realized that as a social consideration variable stakeholder smartphones would inevitably yield variable image dimension and file types, making my assumption of a consistent file type problematic.

This would create dimension mismatches between each of the neural network layers. Researching this particularity I found iPhones capture camera data with jpg, Samsung uses .hiec and google pixel, .raw with output dimensions ranging from 640x480 to 2048x2048.

I immediately thought I would have to conduct some form of image manipulation. However, while on device image manipulation appeared technically feasible I began to doubt the full scope of file type variants, which, in hindsight were extensive and constantly evolving. In addition, when profiling these graphical manipulations I noticed a significant increase in my emulated device's processor use which I would expect to negatively impact battery life. This made me look for a tangible solution. Researching this problem I identified a flaw in my programming knowledge. Originally in fig 27-a the input tensor (`input_size`) was initialised according to my predetermined python constant which was a .jpg batch at 512x512 (`[batch_size, 512, 512, 3]`).

However, when building variable size graphs Jeff stated an important thing to remember is not to encode the batch size as a constant but instead use a symbolic tensor to represent it. This modification would allow me to give my input tensor both a static (inferred) and dynamic (true) shape as later, the dynamic tensor provided leverage to use `Tensor.get_shape`. This method returned a matrix derived from the input image format and dimensions, allowing me to adjust input_size accordingly. This change was fairly straightforward: `input_size = in_layer.get_shape().as_list()`

Since I can expect there will be multiple ingredients for each meal I enumerate over each line of ingredients in my dataset and encode its respective value into its own child input tensor. Similarly, the function dish-name-encoder reads the dish name as specified by its directory and encodes the string on top of the image.

It is important to note that encoding refers to the process of teaching the neural network what different foods look like. Encoding is not static.

During my first *'feed flow'* test, where I pass data through the ingredient-image encoder without actually training, functionality appeared at first normal. I was pleased to see compatibility between a range of image dimensions, however I noticed it performed at below than expected efficiency, indicated by a large delay between input and output. At first I thought this problem was attributed to large test images. However, further debugging revealed that as an image was passed through the network the receptive field of each convolution later was greater than that of each feature map. Breaking down my code I found it's receptive field size was growing exponentially throughout the test. I experimented with passing different sizes through the net but this had no effect for both a 16x16 and 1024x1024 images, thus concluding variable image sizes were not the problem.

Raising this issue with Jeff he suggested my omnitance of *'pooling layers'* were likely a probable cause to inefficiency. Jeff explained pooling layers downsample the image data extracted by the convolutional layers to reduce the dimensionality of the feature map, ultimately decreasing processing time. Obviously this is in the interest of my stakeholders and an important consideration for my brief's specification of efficiency.

A commonly used pooling algorithm is max pooling, which extracts subregions of the feature map (e.g, 2x2-pixel tiles), keeps their maximum value, and discards all others. Extrapolating from other practices I found it is common to periodically insert a pooling layer in-between successive convolutional layers (ConvLs) in a CNN architecture such as mine. Pooling is also aimed at controlling the aforementioned complexity of overfitting[19].

However, reducing computations through pooling too much increases the training time proportionality. It is obvious that too much pooling may be destructive so an important question is how many pooling layers are needed for a given image classification problem. Fig 28 shows my first attempt at this implementation which proved to be problematic.

Pooling Layer Implementation
Fig 28

```
def att_pooling_layer(self, in_layer):
        return tf.nn.max_pool(in_layer, ksize=[1, 2, 2, 1],
strides=[1, 2, 2, 1], padding='SAME')
```

Outcome:

```
ValueError: ('filter must not be larger than the input: ', 'Filter:
[', Dimension(2), 'x', Dimension(2), '] ', 'Input: [', Dimension(1),
'x', Dimension(2), '] ')
```

After much research I realised the error had to do with my input data shape. Whilst Fig 28 was syntactically correct the error arose when my `[1,2,2,1]` tensor was sent to one of the convolution layers which expected an input size of `[1,1,1,1]`.

This mismatch in dimensions pertaining to k-size and strides needed to be compatible with both the training tensor and the output. After understanding this I decided to make all layer sizes constant to prevent further errors.

Loading Input Data for Training

With a technically functional neural network I then began working on the low level functions which would handle loading data during training. A key issue I encountered early on was making this process as efficient

[19] "A Fast Learning Algorithm for Deep Belief Nets - University of Toronto ...." http://www.cs.toronto.edu/~fritz/absps/ncfast.pdf. Accessed 4 June. 2018.

as possible. This was key to minimize training duration yet particularly challenging given the scale of collected data I was attempting working with.

To load training data I had identified I would store information within a list/array data format with each item having its own name, ingredients and images. This practice was in accordance with Google's machine learning crash course notes[20], and from past experience I had found Python's numpy arrays perfect for this task.

Lineary Loading Images Into Memory For Training
Fig 29 | Source: Self

```
for fpath in fpaths:
  img = cv2.imread(fpath,
cv2.CV_LOAD_IMAGE_COLOR)
  image_list.append(img)
data = numpy.vstack(image_list)
data = data.reshape(-1, 512, 512, 3)
data = data.transpose(0, 3, 1, 2)
```

My first attempt is illustrated in fig 29. Although this technique is functional on small scale tests I quickly identified several superfluous operations were being performed. At its present state, I would initialize, reshape, and then transpose each image into a float array. Since images were loaded via a sequential stack arranged vertically (via a *vstack*) Numpy also made a temporary copy of the data in memory. The detriments of this approach became clear during a memory stress test with a larger slice of my dataset.

After research this complexity I encountered Hendric J. Weideman's (PhD candidate)[21] experiments regarding efficient image loading. He pointed out that since we will generally know the number of images we wish to load a better method is to initialize the data array before the transposition of images. I synthesised this new technique on top of my precursive attempt, illustrated in fig 30.

For testing purposes I loaded in 11260 images (of image resolution 256x256) into an empty 8 bit array with dimensions of 256x256, it's worth noting that I do not transpose any entries at this point. I then enumerate over my image directories as they appear on my local file system inserting each as a transposed object with respect to image channel, width and height into the data array.

Through functional modeling I found the improvement Weideman's technique posed over mine was significant, and facilitated an average improvement was 3-4 seconds faster per image (shown in fig 31-b).

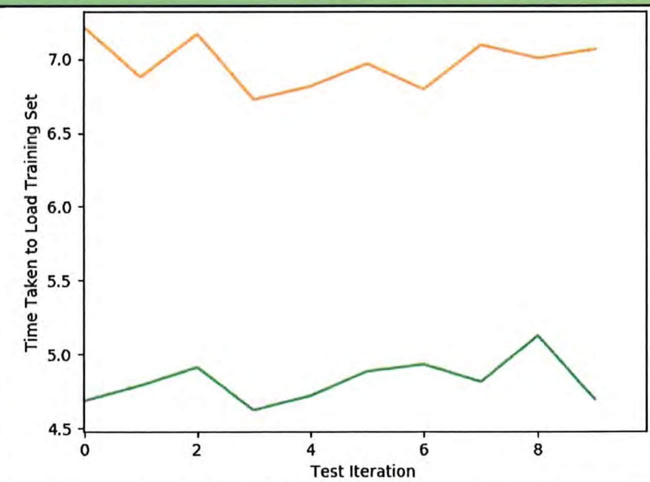
Pre-Initialised Sequential Image Loading
Fig 30 | Source: Self

```
data = numpy.empty((N, 3, 256,
256),
dtype = np.uint8)

for i, fpath in enumerate(fpaths):
    img = cv2.imread(fpath,
v2.CV_LOAD_IMAGE_COLOR)

data[i,...] = img.transpose(2,0,1)
```

Functional Modeling of Image Loading Performance
Fig 31 | Source: Self
(Green line (bottom) = Efficient Method) (Orange Line (Top) = Slow Method)



However, when testing these improvements on my entire dataset training performance was still diminished. I decided that loading an entire dataset into memory was not a feasible approach. When

[20] "Machine Learning Crash Course - Google Developers." https://developers.google.com/machine-learning/crash-course/ml-intro. Accessed 22 Apr. 2018.

[21] "Efficient Image Loading for Deep Learning · Hendrik J. Weideman." https://hjweide.github.io/efficient-image-loading. Accessed 20 Apr. 2018.

discussing this complexity with Jeff he suggested I should load sections of my dataset one at a time. With this in mind I wrote a function which which would return a set number of images (batches) from my dataset Conceptually speaking these batches would then be fed into the joint *ingredient-image* encoders and then be disposed of appropriately once no longer needed. This would allow me to regulate the number of images my training algorithm's memory. My first attempt at managing this complexity is illustrated in fig-32.

Batched Image Generator Implementation
Fig 32

```
def get_sub_batches(self,batch_size=10):
    images, labels = []
    empty=False
    counter=0
 each_batch_size=int(batch_size/len(self.data_info))+1
    while True:
        for i in range(len(self.data_labels)):
          label = np.zeros(len(self.data_labels),dtype=int)
          label[i] = 1
          if len(self.data_info[i]) < counter+1:
              empty=True
              continue
                 empty=False
          img = cv2.imread(self.data_info[i][counter])
          images.append(img)
          labels.append(label)
          counter++
          if empty:break
          if (counter)%each_batch_size == 0:
                      yield  np.array(images,dtype=np.uint8),
np.array(labels,dtype=np.uint8)
             del images, labels
             images=[]
             labels=[]
```

First I calculate `each_batch_size` which is the product of batch_size divided by the number of image directories in my training set. Next for each food group I load the label(`label`) and image (`img`) and append them to the higher scoped `images`, `labels` list. I return a set batch when the modulo of the current iteration equals zero return the current batch.

Learning from this experience in sizing array variables to desired dimensions before inserting my data as well as batched image loading both present significant improvements to how I work with with my training data.

Training

With a preliminary meal analysis network I progressed onto the process of training. An initial complexity was knowing how many training cycles to conduct. I spoke about this with Jeff but he said there is really no optimum number since it depends of a vast set of parameters unique to the context.

He said to start small and build up. Since this would be only an initial test I elected to train on 170 food groups from my total dataset. This was because from past experience I knew training on my full dataset would be a very time consuming process, taking around 1 - 2 weeks to complete. I did this to ensure that if anything did go wrong I would not have to restart the entire training cycle.

Testing Train functionality
Fig 33

Outcome:

```
CRITICAL:tensorflow:Label banana split has no images in the category training.
Traceback (most recent call last):
   File "scripts/model.py", line 23, in get_image_path
         mod_index = index % len(category_list)
ZeroDivisionError: integer division or modulo by zero
```

A key problem which I faced early on was understanding why the neural network was throwing a ZeroDivsionError for a different food group each time I ran the training process. According to fig 33 the stacktrace revealed the food group '*banana split*' had no images in the category training. Upon inspection of the group directly I found that there were 19 images. Although this number was below optimum and raised the question of resultant accuracy (which I will address below) it did not answer the initial error. What was especially confusing was this group with 'no images' would change every time. Logically I would expect my program to iterate sequentially over each group as it were structured in my file system, meaning that if they contained no images then the same group would be specified each time the trace was thrown. However, this was not the case. After extensive research I realised that Tensorflow puts images into a training- testing - validation bucket based on a hash of an images filename.

Test of ingredient and label encoders on new unseen meal image
Fig 34

```
graph = load_graph(model_file)
t = read_tensor_from_image_file(file_name)
with tf.Session(graph=graph) as sess:
   start = time.time()
                                        results          =
sess.run(output_operation.outputs[0],
{input_operation.outputs[0]: t})
   end = time.time()
   results = np.squeeze(results)
   top_k = results()
   labels = load_labels()
   ingredients = load_ingredients()
   test_result = "{} (confidence={:0.5f})"
print(test_result.format(labels[i],
results[i], ingredients[i]))
```

Input:

Output:

```
['double tomato bruschetta']
(confidence=0.6160)
['(plum)', 'tomatoes,', 'chopped']
['sun-dried', 'tomatoes,', 'packed', 'in', 'oil']
['minced', 'garlic']
['olive', 'oil']
['balsamic', 'vinegar']
['fresh', 'basil,', 'stems', 'removed']
['salt']
['ground', 'black', 'pepper']
['baguette']
['shredded', 'mozzarella', 'cheese']
```

This alluded to the apparent random nature of the food group error. It occured that one bucket could be empty for an entire category if the total number of images could not be divided evenly. In my instance 19 images within the 'banana split' and closer inspection of other error cases pertained to this criteria. This pointed to a flaw in my dataset rather than the program itself and made me rethink food data web scraper collection and storage.

Reflecting back to the aforementioned concern of granularity with the '*banana split*' error I began to doubt each food group in my training set as an adequate representation of that food in the real world. This representation is integral to successfully training my neural network. When discussing this problem with Jeff he suggested each group should contain at least 150 images as this would ensure reasonable examples of differing contexts. With these modifications in mind I regenerated my whole dataset making sure to only include meals from Allrecipes.com pertaining to this criteria. Despite initially eliminating a large number of food groups, and therefore negatively compromising on granularity, I was able to counter this by increasing the web scraper's *'depth'* variable proportionally.

To test everything was working correctly I created a debugging block used to demonstrate my first meal ingredient inference in Fig-34 bottom. The test image was of some bread and tomato slices made at home, ensuring to take a picture of my own food so the prototype's performance was a true representation of what it had learned, eliminating the potential for overfitting (if present) to distort my results. The meal also contained one of many types of bread (sliced baguette) as well as specific ingredients like tomato. This would be a good way to gauge its ability to abstract accuracy. The test was successful and I evaluate it in full in the preceding section. Note how figure 34 shows output in a raw format. I did this to illustrate bare functionality without an interface and will no doubt raise social consideration concerns when evaluated against stakeholder feedback.

I also carried out a range of tests on things I had eaten Fig-35 and reference these in my evaluation.

Full Test of Cross Neural Network Embeddings on unseen real world images.
Fig 35 | Source: self

TEST 1

```
Evaluation time: 2.315s
Name: bananas foster ii
Confidence: 0.5227
['butter']
['dark', 'brown', 'sugar']
['beer', 'rum']
['teaspoons', 'vanilla', 'extract']
['ground', 'cinnamon']
['coarsely', 'chopped', 'walnuts']
['vanilla', 'ice', 'cream']
```

TEST 2

```
Evaluation time: 1.117s
Name: double tomato bruschetta
Confidence: 0.11735)
['(plum)', 'tomatoes,', 'chopped']
['sun-dried', 'tomatoes,', 'packed',
'in', 'oil']
['minced', 'garlic']
['olive', 'oil']
['balsamic', 'vinegar']
['fresh', 'basil,', 'stems', 're-
moved']
['salt']
['ground', 'black', 'pepper']
['baguette']
['shredded', 'mozzarella', 'cheese']
```

TEST 3

```
Evaluation time: 1.034s
Name: old charleston style omlett
Confidence: 0.6627
['cayenne', 'pepper']
['andouille', 'sausage,', 'cut']
['bacon']
['bell', 'pepper,', 'chopped']
['chopped', 'onion']
['minced', 'garlic']
['butter']
['all-purpose', 'flour']
['chicken', 'broth']
['Worcestershire', 'sauce']
['shredded', 'sharp', 'Cheddar',
'cheese']
```

TEST 4

```
Evaluation time: 1.049s
Name: turkey burger
Confidence: 0.18893
['ground', 'turkey']
['seasoned', 'bread', 'crumbs']
['finely', 'diced', 'onion']
['whites,', 'lightly', 'beaten']
['chopped', 'fresh', 'parsley']
['garlic,', 'peeled', 'and',
'minced']
['salt']
['ground', 'black', 'pepper']
```

Initial Evaluation and Reflection

Stakeholder Evaluation

Discussion with Danijela

I presented the current neural network to Danijela as well as my test findings in Fig 35. I also asked Danijela to provide several of her own images to demonstrate the neural network's performance on, which she did. Danijela was able to understand the prototype's output, despite its raw text format, and was impressed by its initial performance. I noted her liking towards the specificity of ingredients such as *'sugar'* in the *'ice cream'* in fig-35-test1 a and *'basil'* in fig-35-test2. I attributed this specificity in performance to my importance placed on granularity within my dataset, and as an indication of increased fitness for purpose.

However, a particular area she suggested could prevent me from making an accurate inference towards the healthiness of food was the absence of ingredient measurements. I saw this information would also play a wider, more important, role in educating people why their meals were not healthy. Danijela said measurements are integral to quantifying the health benefits of a meal. For example, knowing the amount of flour used in a cake is paramount to calculating its calories. This would also be necessary in facilitating the *ideal meal portion ratio* as a way to compare stakeholders against an ideal case.

I noted, as a relevant social consideration, the metrics for these calculated ingredients. In countries where the imperial system is used, *ie United States*, people might be confused if my meal ingredients used the metric system. Conversely people from NZ might struggle to understand ounce to gram conversions, and having to do so could hinder efficiency within time pressured contexts. Converting between the two will become an important future consideration, necessary to ensure global scalability.

Whilst the neural network's successful inference of ingredient keywords is a necessary step in addressing my identified issue, the output data shown in Fig-35 does not pertain closely to my projects value of education. It occurred to me that simply stating ingredients of food situates my solution no better than the already in use nutritional summaries on food packaging, failing to inform people *why* their meal is unhealthy. I therefore need to manipulate generated ingredient data in such a way to draw a meaningful conclusion, determining whether it is detrimental to someone's health.

Furthermore, my findings from the client re-visitation only imply that nutritionists, represented by Danijela, are able to interpret the output. I fear that it was not an accurate reflection of how all potential users of my solution would react when interacting with the neural network. Since I had consulted Danijela throughout the production of the prototype she was well aware of its functionality. Additionally Danijela had a formal university education and already knew what to look for when analysing food. Therefor, as outlined in my brief a key value of my project is to be scalable so assuming the status quo of my wider stakeholders mirrors that of Danijela would be problematic.

Because of this, my evaluation is likely not representative of a trend more to do with less motivated people and those inexperienced in interpreting nutritional data. My direct input in taking Danijelas images and feeding them to the neural network means this test was also not reflective of a real world situation. There was no element of time pressure nor severe health consequences if the output was incorrect. After my discussion with Danijela her feedback affirmed my established importance of situating my eventual solution within a mobile app.

Reflection

Considering the feedback from Danijela and my analysis of its validity I belief that my project is not fit for purpose in the broadest sense. The lack of user interface, inability to infer why a meal is unhealthy and deriving ingredient measurements all present several important improvements which need to be made to ensure compliance with my brief.

Furthermore, lessons learned from this discussion will be applied to my final evaluation in ensuring the test of my final prototype is fully representative of all my stakeholder groups. This will allow me to definitely determine whether my project is is fit for purpose with true regard to the physical and social context.

Independent Evaluation and Reflection

Functional Modeling of Neural Network

It is also equally relevant that I evaluate the neural network's technical performance. My four initial test cases in Fig-35 left me initially concerned regarding their low confidence scores, none of them were above 0.68 (68%) accurate. When I graphed training accuracy with respect to the training cycle I was able to evaluate this in full.

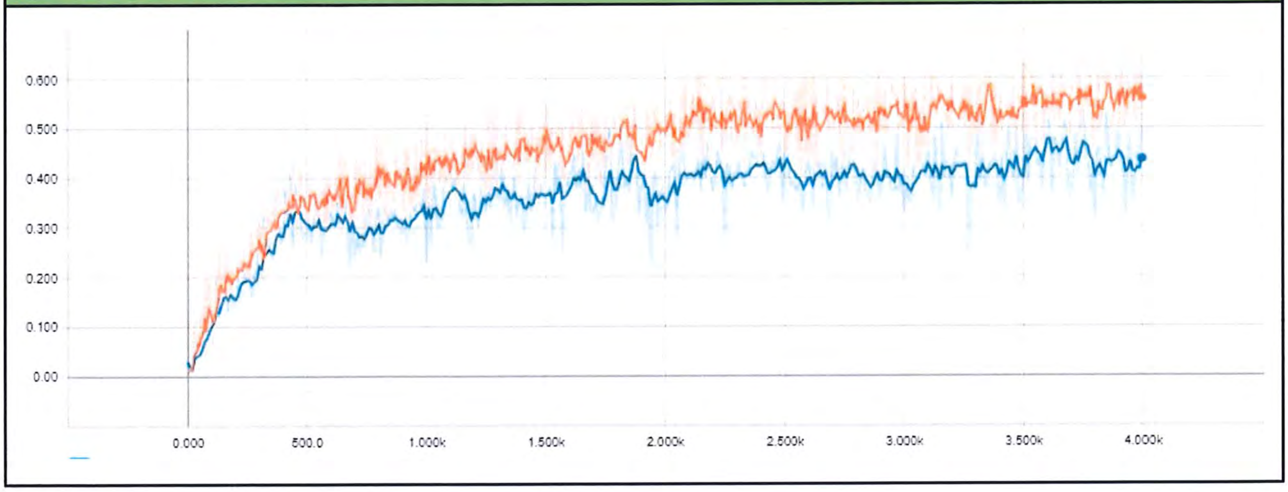
Fig-36 (next page) shows the percentage of images used in each training batch that were classified with the correct dish name and ingredients. The blue and orange lines represent the averaged accuracy when testing on *validation* and *testing* datasets respectively. I will consider the blue line as worst case as it contained images the neural network had not encountered and in that way better represents stakeholder use in an unfamiliar stakeholder's physical environment.

From this graph I found that a square root relationship existed between training cycles and accuracy. During my training period of two weeks none of the test cases evaluated past 50% accuracy, less than the Fig-35 tests!. Not only do these results fundamentally go against this projects value of accuracy it also does not invoke a strong emotional resonance as it appears more often that not my prototype would be misleading, rather than informing its users. Furthermore, this test was carried out on my laptop so it does not reflect true performance had the network been situated in a mobile environment and required to conform to its computational envelope.

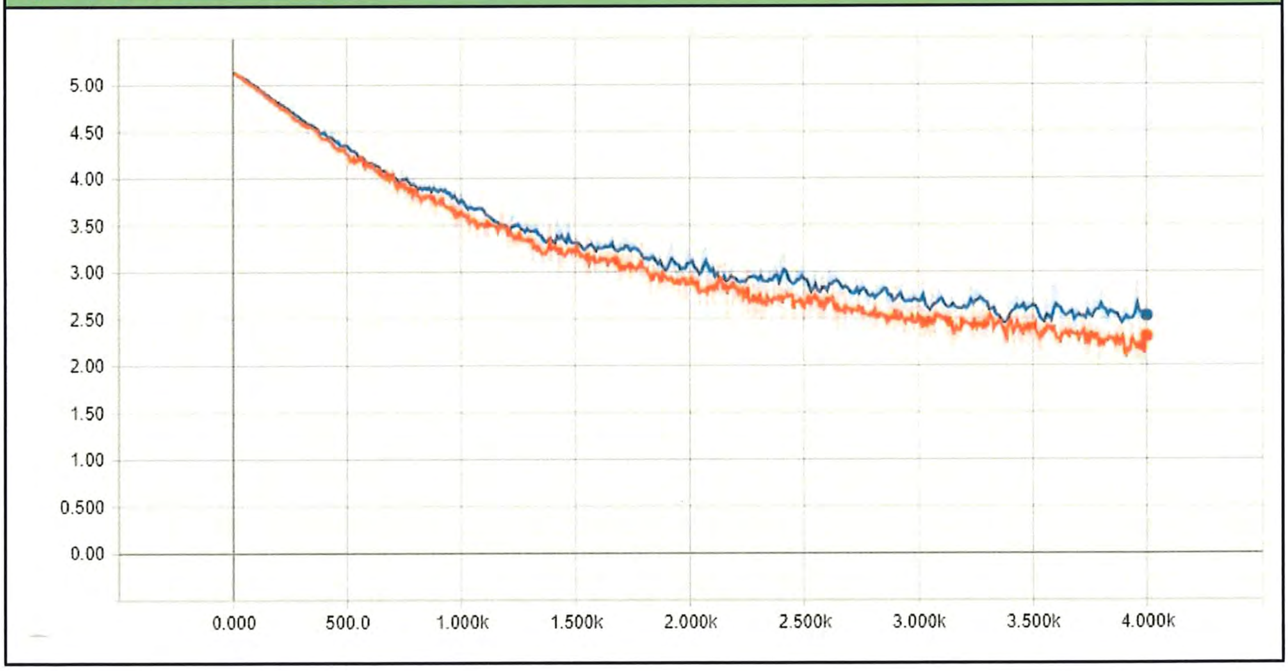
However, a positive trend in accuracy is present past 4.0k cycles, and given an square relationship will always continue increasing, perhaps the simplest means of improving accuracy would be to let the network train for longer, around six weeks rather than the previous two. I was cautious to do this because when talking to Jeff training on a fixed dataset for a extend period can lead to the network overfitting training data. Jeff pointed out to safely maintain true accuracy through a extended training cycle I must increase my dataset proportionally. I knew I could this efficiently by increasing my web scraper depth variable.

Another important point of evaluation is cross entropy loss. Cross entropy is a loss function that gives a indication into how well the learning process is progressing and measures the performance of the classification encoder's learning rate in terms of correct predictions. A perfect model would have a CEL of 0 by the end of training but by the end of Fig-37 the lowest loss value is 2.51. This indicates a significant performance deficiency, evident in my evaluation and testing with Danijela as often I had to pass the image through the network several times before it correctly made the prediction.

Graph of Prediction Accuracy vs Number of Training Cycles
Fig 36 | Source: self



Graph of Cross Entropy Loss vs Training Cycle
Fig 37 | Source: self



Training accuracy and Cross Entropy Loss both give a good indication of the neural networks performance and both point to significant deficines in accuracy and learning rate.

Based on my evaluation results I feel that my prototype is not fit for physical purpose due to its poor performance in both inference accuracy and CEL, let alone the low-power mobile devices my project must also be fit for purpose on to fulfil my importance placed on portability. Therefore, improving accuracy will be another major focus in the production of my final prototype. Improvements will also be made as to how the social fitness for purpose can be improved based on my analysis of stakeholder feedback.

Final Prototype Development

After analysing my stakeholder feedback, self reflection and performance investigation I feel that I have a very good sense of what the major limitations of my project are in terms of both the physical and social contexts. I believe that the performance issues are the most significant as the accuracy trend during testing proved I was more often misleading stakeholders than not. This contributes to the broader issue of a general lack of user friendliness in the solution, attributed also by a lack of UI and means of conveying meal 'healthiness'. Compromising the solutions ability to present information clearly and address the issue.

Improving the Neural Network

Synthesis of Evaluation Findings into Conceptualisation

The main conclusion I drew from my performance investigation was that minimal accuracy combined with not being able to derive quantitative measurement data from pictures of food was both counter intuitive to addressing nutrition misinformation and preventing an overall inference of meal healthiness being made. I can attribute this poor performance to be linked to the makeup of my training data due to its close correlation, and, my existing implementation of joint neural embeddings (Fig-26) for each modality of food nutrition information in the neural network.

Currently, under the joint embedding approach, only ingredient key words, the meal name and images are used as traininable example pairs, with convolution applied when text is inferred from visual image information. The logical approach would be to add another module encoder to facilitate ingredient calculation. However, from past experience training a NN to a high degree of output modality, ie, *name*, *ingredients* and *measurements*, this can often prove detrimental to efficiency as more calculations are required per inference.

However, from my evaluation tests both Danijela and I regarded efficiency as acceptable with each classification only taking on average one to two seconds. I would expect the benefits of adding a seperate encoder designed for measurement-ingredient inferences on top of the existing image-ingredient one to outweigh the detriments in marginal compromises to the user experience it would pose. If present these detriments can be mitigated through optimisation techniques Jeff highlighted at the start of this report (they are and discussed/implemented below).

Obviously this approach would require I first gather ingredient specific measurement data and add it to my ingredient-image training pairs. However, to make use of this new data I will need to convert conventional recipe measurements to percentages relative to their total presence in a meal. This is important in inferring the prevalence of ingredients and ultimately making a judgement of the overall healthiness

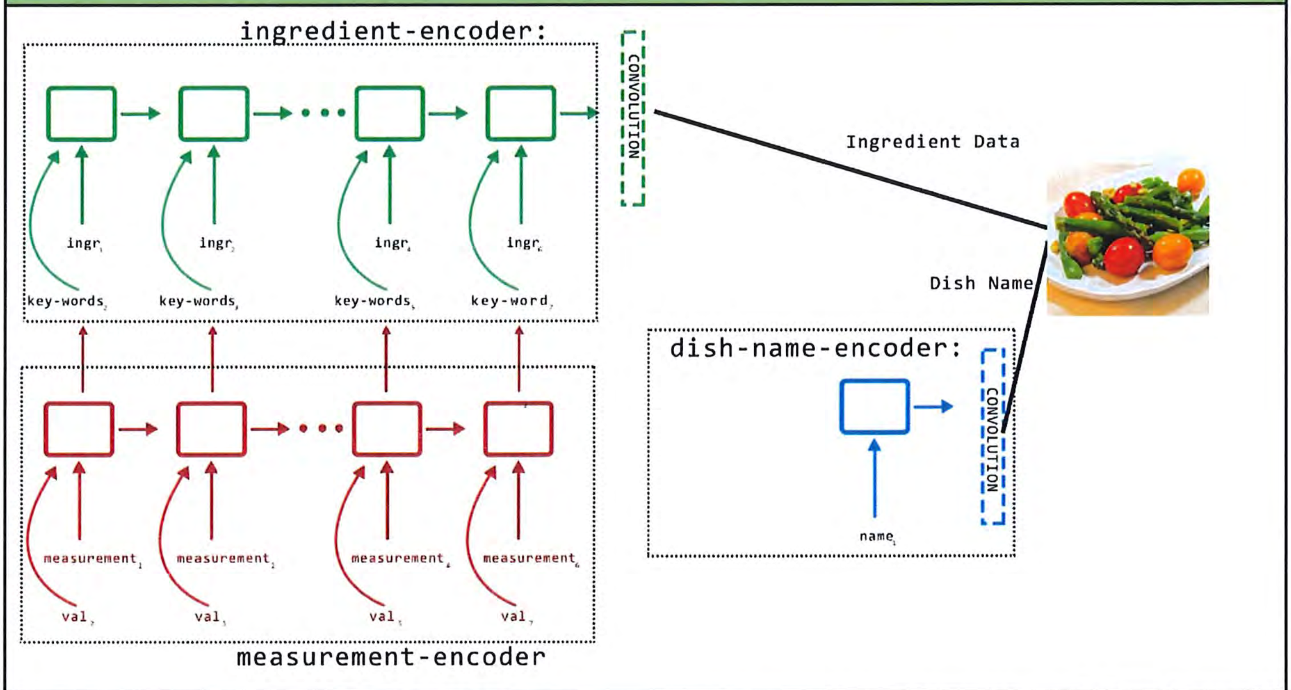
In using a percentage I can also easily translate this data into a graph to show the *ideal meal portion ratio*, and, to an extent mitigate the social consideration metric and imperial measurements pose since a percentage in terms of total meal composition is independent of a unit to which its recipe measurements were originally based off.

However, metric to imperial conversions will still be necessary as these percentages will only be used in graph generation rather than displayed in a raw format. This is because I have established large blocks of number percentages discourage stakeholders from reading them.

Improvements to network architecture

I started by remodeling the joint embedding encoders of the neural network to better accommodate measurement inferences. The new measurement-ingredient encoder utilises the same approach in convolutional neural networking as the ingredient-image encoder except it allows for integers in label data, important for taking measurements. The measurement encoder takes ingredient keywords and added this to the training examples pairs which pass through a series of convolution layers before being weighed to their training images.

Improved joint embedding approach to accommodate ingredient measurements.
Fig 38 | Source: self




```
2018-05-29 21:35:11.508893: Step 3930: Train accuracy = 95.0%
2018-05-29 21:35:11.509392: Step 3930: Cross entropy = 0.221186
2018-05-29 21:35:11.671590: Step 3930: Validation accuracy = 93.0% (N=100)
2018-05-29 21:35:13.252652: Step 3940: Train accuracy = 96.0%
2018-05-29 21:35:13.253152: Step 3940: Cross entropy = 0.156158
2018-05-29 21:35:13.410857: Step 3940: Validation accuracy = 98.0% (N=100)
2018-05-29 21:35:14.932031: Step 3950: Train accuracy = 99.0%
2018-05-29 21:35:14.932531: Step 3950: Cross entropy = 0.089506
2018-05-29 21:35:15.096257: Step 3950: Validation accuracy = 91.0% (N=100)
2018-05-29 21:35:16.588483: Step 3960: Train accuracy = 95.0%
2018-05-29 21:35:16.588483: Step 3960: Cross entropy = 0.168742
2018-05-29 21:35:16.752222: Step 3960: Validation accuracy = 94.0% (N=100)
2018-05-29 21:35:18.248940: Step 3970: Train accuracy = 94.0%
2018-05-29 21:35:18.248940: Step 3970: Cross entropy = 0.159020
2018-05-29 21:35:18.404152: Step 3970: Validation accuracy = 95.0% (N=100)
2018-05-29 21:35:19.895381: Step 3980: Train accuracy = 97.0%
2018-05-29 21:35:19.895880: Step 3980: Cross entropy = 0.157969
2018-05-29 21:35:20.060189: Step 3980: Validation accuracy = 91.0% (N=100)
2018-05-29 21:35:21.530955: Step 3990: Train accuracy = 96.0%
2018-05-29 21:35:21.530955: Step 3990: Cross entropy = 0.139345
2018-05-29 21:35:21.679180: Step 3990: Validation accuracy = 84.0% (N=100)
2018-05-29 21:35:23.015696: Step 3999: Train accuracy = 97.0%
2018-05-29 21:35:23.016194: Step 3999: Cross entropy = 0.116835
2018-05-29 21:35:23.181888: Step 3999: Validation accuracy = 84.0% (N=100)
Final test accuracy = 91.3% (N=664)
```

To verify adding the additional encoder module yielded improvements to the output of nutritional information I performed a functional modelling investigation.

To the left I show the training, validation and cross entropy logs during the last few seconds of a five week training block. The end result gives 91% test accuracy. Although the physical and social implications of this will be evaluated in full at the end of this report this is significantly greater than the 46% validation and 53% testing accuracy from my initial evaluation prior.

Improvements to Data Set

Test of Conventional Recipe Measurement to Metric Converter | Fig 40

```
from fractions import Fraction
ingredients = see input below
cup_grams = 340
teaspoon_grams = 4.2
def strFraction(value):
    return float(sum(Fraction(s) for s in
value.split()))
teaspoon,cup,organics =[],[],[]
for line in ingredients.split('\n'):
    if "tea" in line: teaspoon.append(line)
    elif "cup" in line: cup.append(line)
    else: organics.append(line)
print("--Teaspoon Measurements--")
for a in teaspoon:
    vala = strFraction(a.split(' ')[:1][0])
    print (str(vala * teaspoon_grams)+ "g", a.split('
')[2:])
print("\n--Cup Measurements--")
for a in cup:
    val = strFraction(a.split(' ')[:1][0])
    print (str(val * cup_grams) + "g", a.split(' ')[2:])
```

Converting conventional recipe measurements such as cups and teaspoons to the metric system was another important complexity I needed to address after my evaluation. This was particularly challenging due to ingredient data from Allrecipes being downloaded in a raw string format. It would not be technically possible to simply multiply these values by the coefficients needed to convert into metric units as I first hard to localise these numbers within the string. However, inconsistency in all recipe ingredient data made this highly problematic.

After much thought I realised that the problem could be simplified by dividing the ingredients into three categories: *teaspoons*, *cups* and *organics*. The first two categories are self explanatory but the *organics* category encapsulates

```
print("\n--Organics--")
for a in organics: print(a)
```

Input:

```
1 1/2 cups all-purpose flour
1/4 teaspoon baking soda
1/4 teaspoon sea salt
1/2 cup butter, softened
1/4 cup white sugar
1 egg
2 teaspoons vanilla extract
1 cup chocolate chips
```

Output:

```
--Teaspoon Measurements--
  1.05g ['baking', 'soda']
  1.05g ['sea', 'salt']
  8.4g ['vanilla', 'extract']
--Cup Measurements--
  340.0g ['cups', 'all-purpose', 'flour']
  170.0g ['butter,', 'softened']
  85.0g ['white', 'sugar']
  340.0g ['chocolate', 'chips']
--Organics--
  1 egg
```

ingredients who share a unique unit such as *1x apple* or *3x celery sticks* and thus was logical to leave as is. Although Allrecipes automatically formatted meal ingredients in a specific way occasionally there were irregularities due to users including imperial and metric measurements on the same lineline. The organics group caught these cases as well.

In the script to the left I employ this technique and this allow me to identify the numerical measurements in each ingredient category and simply multiply these numbers by the coefficient required to convert them into metric units.

User Interface

The user interface was an important attribute identified to be lacking during my initial evaluation. It will become the means by which my stakeholders interact with the neural network.

Component Outline

At this stage my prototype exists within a command line environment, and whilst technically functional, to ensure I meet my brief specification of usability I must consider the user interface which the neural network will be situated in. When discussing this with Danijela at the beginning of my report and in my initial evaluation I had established the importance of situating my solution within an app, and had gained a preliminaliminary understanding of the required specifications, they are summarized below.

Specificational Recap

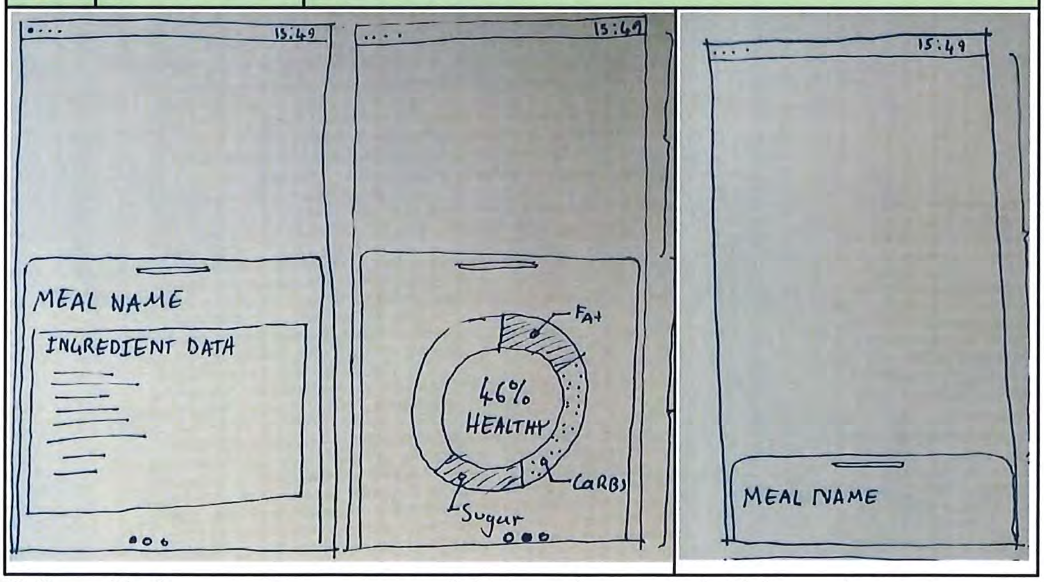
- Readability enabled through a visually and quickly scannable output aesthetic
- Incorporate a Information Hierarchy by means of information separation by importance
- Use of Danijelas ideal meal portion ratio to convey *'healthiness'*

Conceptualisation with Stakeholder

My initial idea behind the interface was to have a '*bottom card*' which would be the primary source of information and could be minimised and maximised but the user. This *card* would be overlayed on top of the output from a device's camera and contain a live feed of what the user's meal most likely name and contents are. In this way when the card is minimised only the predicted meal title would be shown but when it is maximised it would display detailed nutrition and portion ratio information. This would show a direct parallel between the image of food in question and the interpreted data outputted from the neural network and go some way to contributing towards my value of user education. My initial wireframing concepts are shown below.

Synthesis and Reflection

Fig 41	Source: self	*First Conceptual Interface*



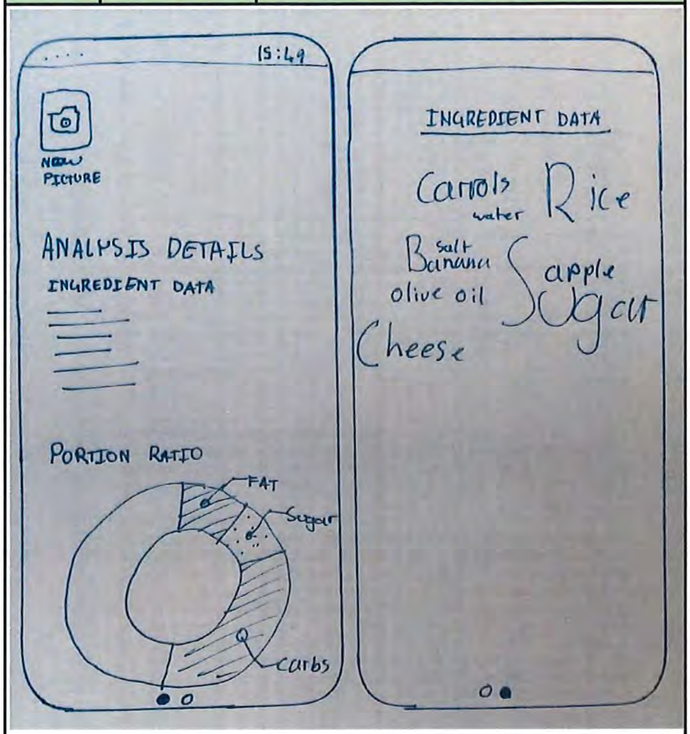
I first showed the designs to my primary stakeholder Danijela and explained the intended functionality behind the interface. She quickly questioned the way ingredient data would be displayed. Going with the current raw output of the neural network (which was just text containing key ingredients - Fig 34) both of us agreed it did not have a quickly scannable aesthetic and made me rethink how I was to convey nutrition information clearly.

The format to which neural network classification is outputted remains closely related to interface design, so from this point I decided to separate Danijela feedback between the format of the neural network output, and the interface itself. In terms of the interface, Danijela suggested I adopt a form information hierarchy to separate information of importance, saying for each analysis case I should first display quickly readable and easily understandable information such as the meal name and a calculated portion ratio which would give people a easily understandable overview of their meal. I saw this would be of benefit for clients using the app in a time pressured context such as those (Chipotles) examined at the start of this report. Danijela said if users wanted to find out more they could scroll down or swipe left to view a more detailed nutritional makeup summary or potentially ingredient data.

However, I had already identified through my survey at the start of this report that they discourage people from reading them due to their complicated aesthetic. This made me think about better ways to convey this information. One which sounded promising was through a *word cloud.* This is a jumble of text that uses scale as a way of weighing the importance of words, or in my case the prevalence of an ingredient in a users meal.

Synthesis and Reflection

Fig 42	Source: self	*Second Conceptual Interface*



Considering this feedback I refined my approach .Fig-42-right shows how I could potentially use a *word cloud* to create a quickly scannable way in showing the prevalence of ingredients in a meal. Larger words would represent a larger presence smaller words less presence. Fig42-left shows what a detailed analysis and portion ratio would look like, previously discussed and synthesized into my concept. This information would be shown secondly only if the user wanted more information.

I showed this revised interface (Fig-32) to Danijela for her thoughts. She pointed out that overlaying analysis data as I had done in the previous concept, made possible through the *bottom card*, was beneficial as it showed parallels between data and food, I agreed with her. But in this concept's layout, I lost that parallel since meal data and meal images are handled between two separate screens. However, Danijela liked my use of the word cloud to represent which ingredients were prevalent and agreed that this would only be effective if the neural network was able to work out the quantities of ingredients.

Reflecting on this feedback I elected to go with the bottom sheet idea but take with me the use of the portion ratio graph to show the prevalence of ingredients. I would also employ the use of a information hierarchy to separate information in order of importance. These attributes from my discussion with Danijela can more effectively facilitate my designs specifications in accordance with a visually scannable aesthetic and ensure quick understandability within time pressured contexts as important attributes within my brief.

Running the Neural Network on a mobile device

Component Outline

Interacting with the neural network in its current state requires an image be passed in normalised float array through a specified directory into the Neural Network. This is obviously not practical for everyday use so as an important future consideration the app must bridge this gap between human and machine interaction.

Tool and Material Selection

The first step to developing the app was to select an appropriate IDE to enable me to design the app and publish it to my stakeholders devices.

Android Studio: After discussing what I needed my conceptualised app to do with Jeff he suggested I start by looking into Android Studio. Jeff used Android Studio hisemf and said that it was a purpose-built suite of tools for enabling android specific app development. Despite being industry standard, I found it went against my project value of scalability as isolating my solution to android users only would reduce my outcome's overall fitness for purpose.

Xamarin: Another possible IDE I could use was Xamarin. I had used Xamarin in the past and was fairly familiar with it. An obvious benefit it held was being able to target all mobile platforms through a single shared codebase for android, ios, and windows. In this way I write one app and Xamarin automatically builds to each mobile os. This would ensure all stakeholders, regardless of their phone type, would be able to use the app and no doubt guarantee usability in accordance with my brief. However, as I researched deeper into this platform I realised due to major differences in the Android and iOS architecture transferable code was limited to predominantly high level methods. This was only great for code designed to manipulate an interface. I raised this with Jeff who said often developers ended up having to write platform specific code for low level device operations. Since I knew I needed to interface with the camera and hardware level features this created an immediate detriment and somewhat eliminated the benefit a dynamic codebase would create in the first place.

Tool Selection Reflection

While the potential benefits of Xamarin do to some extent allow for simultaneous cross device development I decided the benefit would be minimal as low level functions would still need to be written under device specific terms. Given these terms I chose to select android studio as my app development IDE.

Interpreting Device Camera Data

I started with the goal of collecting an image from the devices camera and preparing it to be passed into the neural network. A key issue that continually came up was managing the correct permissions associated with programmatically accessing the camera. Android by default requires apps to attain this permission so before interaction so before I could manipulate this data I had to gain the users permission.

Fig 43	Source: self	Implementation of Device Permission Requests

```
@Override public void onRequestPermissionsResult(
   final int requestCode,
   final String[] permissions,
   final int[] grantResults) {
  switch (requestCode) {
   case PERMISSIONS_REQUEST: {
     if (grantResults.length > 0
             && grantResults[0] ==PackageManager.PERMISSION_GRANTED
             && grantResults[1] == PackageManager.PERMISSION_GRANTED){
       //permission obtained, continue with startup
     } else {
       requestPermission();
     }}}}
```

Now that a reliable and convenient method in retrieving user permissions had been developed, I began within the methods which would take camera information and pass it into the neural network. Since image data from an Android device is stored in a 0-255 integer array a key issue I initially ran into was being able to preprocess this data into a normalized float array which the network could interpret

Further researching this complexity showed that android devices also output image data at either 8 or 16 bit color depth depending on the model, further complicating conversions. However, due to the computationally intensive nature of image processing introducing a potentially energy draining operation on camera data is unattractive as it would shorten the device's battery span and have to be repeated for each analysis case.

This would be undesirable and led me to consider whether it would technically and socially feasible to use this approach. When evaluating whether this was possible I was led to a Stack Overflow discussion in which a android developer had a similar issue. One commenter suggested instead of trying to convert an image as you normally would it is possible to specify the view mode of an android's camera to a specific output format, eliminating the need for graphical manipulations in the first place. I therefore modified my process accordingly. While this technique worked to an extent in getting the correct file type and resolution it was still necessary to perform a conversion into the normalised float array format as android does not support this natively.

Fig 44	Source: self	*Converting Image data to a normalized float array*

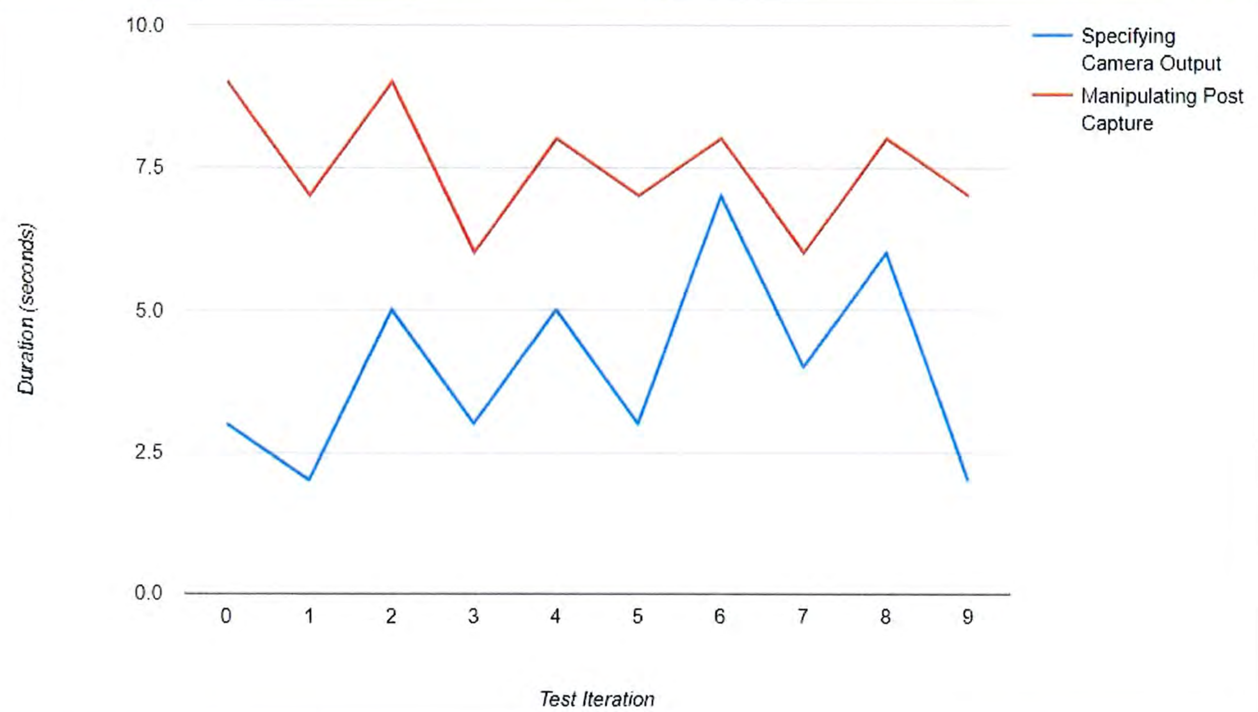
```
intValues = new int[inputSize * inputSize];
bitmap.getPixels(intValues, 0, bitmap.getWidth(), 0, 0,
bitmap.getWidth(), bitmap.getHeight());
for (int i = 0; i < intValues.length; ++i) {
 final int val = intValues[i];
 floatValues[i * 3 + 0] = (((val >> 16) & 0xFF) - imageMean) /
imageStd;
 floatValues[i * 3 + 1] = (((val >> 8) & 0xFF) - imageMean) /
imageStd;
floatValues[i * 3 + 2] = ((val & 0xFF) - imageMean) / imageStd;}
```

Output (shortened for simplicity):

```
[0.8368016,...,0.57148737,0.76597935,0.09562139,0.18201722,0.50720257,0.06972256,0.044948604,0.3042742,0.72387046,0.18761188,0.016413873,0.32732365,0.25570783,0.31156695,0.32365316,0.04827165]
```

To solve this complexity with the raw bitmap data, for each RGB color channel, (represented by a new index within the int array) if the value is greater than 16 and the hexadecimal literal 0xFF of value 255, subtract the mean image dimensions divided by the standard image dimensions. I tested these techniques and graphed the results in fig 45 through functional modeling to evaluate this decision in full.

Fig 45:	Source: self	Functional Modeling of mobile device image collection techniques



This approach allowed me to efficiently interpolate int values from 0 to 255 into a 32 bit float format which the NN could interpret and was significantly faster than getting camera data and manipulating it using conventional process techniques on device. Improving this areas fitness for purpose and contributing towards my importance placed on efficiency.

Mobile Neural Network Embeddings

Now that a functional method had been developed to interpret data from the camera I progressed onto embedding the neural network into the mobile app. I discussed this part of my project with Jeff who immediately pointed out the importance of preparing the network for a mobile environment. He surmised the process into two stages, optimisation and quantization.

Optimisation

Jeff said the biggest difference between running neural networks with mobile and desktop environments is the importance of its binary file size. On desktop machines it's not unusual to have neural networks which are hundreds of megabytes, but for mobile this is not possible due to physical constraints. This raised an important social environment consideration. A small size is beneficial as minimal data would be used to initially download the app. Obviously, this is of interest to people with cheaper smartphones with less space and cannot afford to download a large app over a internet connection.

Jeff stated TensorFlow included a subset of standard optimization implementations by default which can generally guarantee around a 12 MB final network file. Although I was able to achieve acceptable files size through these ops alone I wanted to reduce this further. Additional optimization allowed me to eliminate libraries which only included implementations of the ops actually required, since throughout development I had experimented with quite a few I never used. This process was fairly straight forward and shown below.

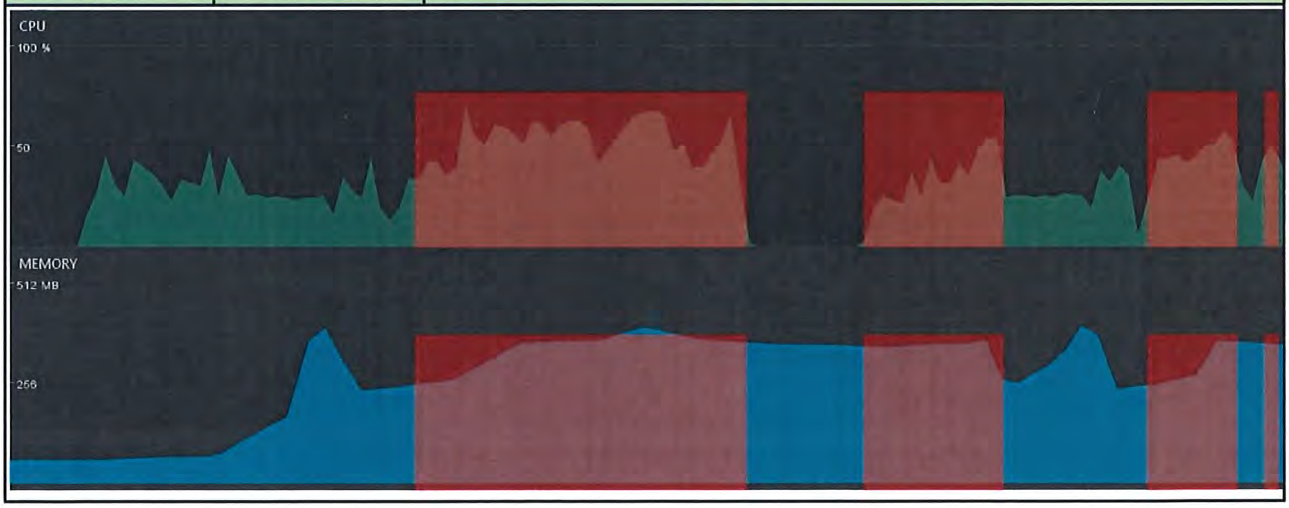
Quantization

After some time I realised a majority of the space taken up by the neural network was by its weight values, which are large blocks of floating point numbers derived during training, each containing slightly different values. I considered rounding them but Jeff discouraged this as these values had been fine tuned during the training process and in changing them I risked distorting the neural networks predictions. A common trend was these values shared very little regularity, making it hard to find a pattern that could be simplified. Since floating point numbers take up more space than rounded floats or integers converting them is desirable. The solution was quantization which is a process that constrains an input from a continuous or otherwise large set of values (such as the real numbers) to a discrete set (such as the integers).

Multi-Threading

When I tested the user interface I realised when the neural net was making an inference on an image my device's responsiveness decreased significantly. Since both the user interface and neural network were running on the same thread this left the device frozen for 1-2 seconds seconds.

Fig 46: | Source: self | Mobile Device CPU (top) and Memory Usage (bottom) Functional Mode



I profiled both the CPU and memory usage to further understand this. In Fig-46 the red highlighted sections indicate when the NN was making an inference and CPU usage (green) and memory (blue) both spike during this time, although more evident within the CPU profile.

This suggests running both classification and UI process on the same thread is derogatory to the user experience and while a brief delay to make a classification is unavoidable it should not impact on a device's responsiveness. At the moment this does not invoke emotional resonance. To manage this I turned to multi-threading optimisation. Multi-threading shares the process resources across different threads, allowing them to be executed independently.

Fig 47	Source: self	Multi-threading Implementation

```
runInBackground(
  new Runnable() {
     @Override
       public void run() {
          final List<Classifier.Recognition>
results =
classifier.recognizeImage(croppedBitmap);
...
}}}
```

Application's responsibilities can be separated so the main thread can run the user interface while slow tasks can be sent to background threads.This approach would be particularly useful in the case of a simple process which can spawn a delegation of tasks across multiple threads on top of a multiprocessor system, allowing parallel computing to be achieved. I implemented this ideology in my approach shown in Fig-37.

The first test of running the neural network within a mobile environment is shown below. The input image is passed to the neural net and output data is displayed in Android Studio Logcat (a specialised debugging console attached to the app process on my phone). I took a screenshot with the respective input image alongside. The total evaluation time was 2.334 seconds, only around 1 second slower than my desktop environment tests in fig 16, despite being run on a mobile device with significantly less processing power.

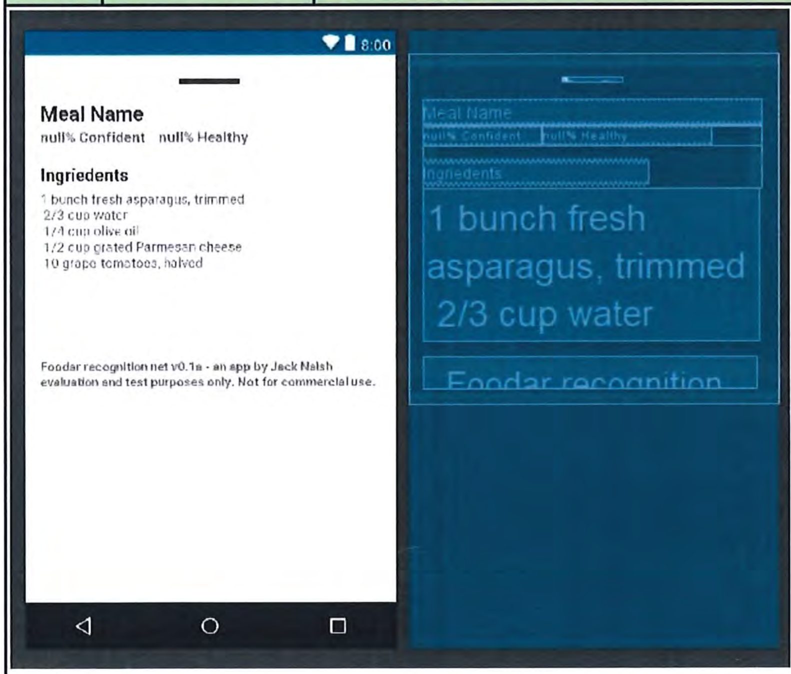
Fig: 48	Source: self	Calculating Meal Ingredients within a mobile environment

```
apple crisp ii (score=0.05947)
10 cups all-purpose apples, peeled, cored and sliced
1 cup white sugar
1 tablespoon all-purpose flour
1 teaspoon ground cinnamon
1/2 cup water
1 cup quick-cooking oats
1 cup all-purpose flour
1 cup packed brown sugar
1/4 teaspoon baking powder
1/4 teaspoon baking soda
1/2 cup butter, melted

Process finished with exit code 0
```

Interface Implementation

Fig 49	Source: self	*First Implementation of Interface*



I then implemented the interface to display the output of the neural network on the screen of a mobile device. This design was based on my research with Danijela in the preceding section. Note it contains placeholder nutrition information.

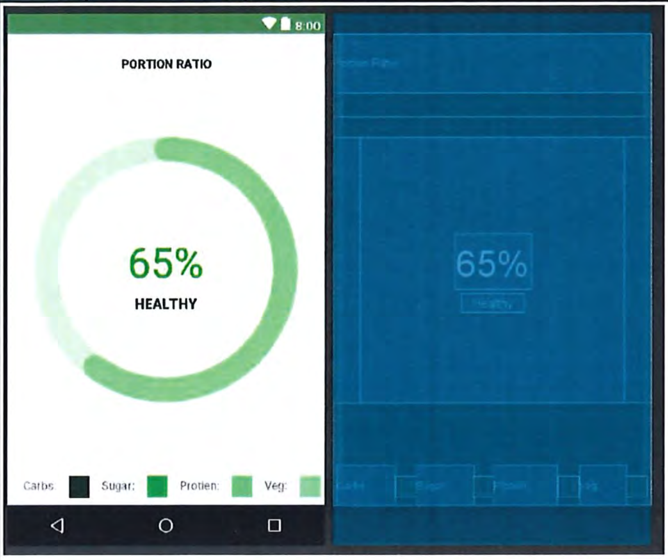
Classification Noise

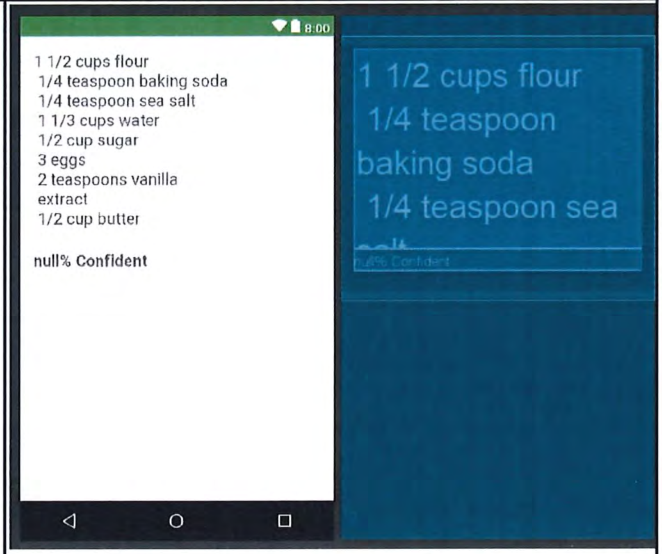
Since the neural network was analyzing every frame provided to it from the device camera this led to a flurry of potential meal classifications. This can be seen happening here[22]. I had not expected this to be a problem since during testing in my pc environment I was working on still images. However, in situ often meal data would flash briefly on the screen before being replaced with that of a the neural net though to be a better option.

[22] "First Interface Test - YouTube." 7 Oct. 2018, https://www.youtube.com/watch?v=jNo4JqsmeM8 30 Oct. 2018.

This made output mostly unreadable. After much thought I decided to implement a threshold accuracy value, of which only classifications with a score greater than would be displayed. I them implemented the user interface in android studio based off my hand drawn interface conceptualisation with Danijela.

Fig 50	Source: self	*Implementation of portion ratio View*	*Implementation of Ingredient View*

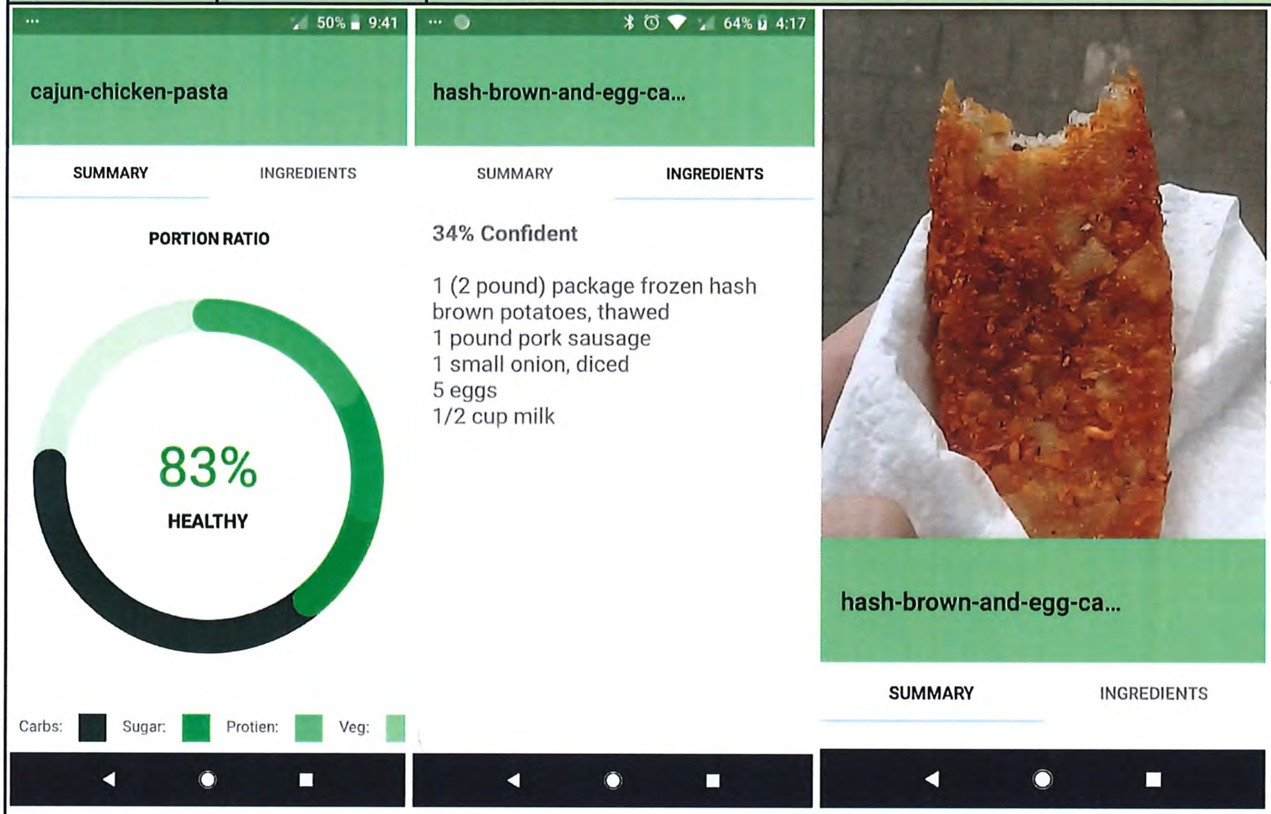




Outcome

Fig-51 demonstrates the full implementation test of my final prototype. In these screenshots a picture of a hashbrown I was eating at school has just been taken and the meal name, portion ratio and ingredients are displayed on screen. The bottom card has been minimised in the right image and maximised in the two left hand images, first showing a quickly scannable graph of Daniela's meal ratio and then a detailed nutritional summary. These tests were conducted in flight mode demonstrating offline performance.

Fig 51:	Source: self	The outcome of the Prototype with me testing on a hashbrown



Video Demonstrations

Portion Ratio Graph Demonstration[23]

Pasta Meal Demonstration [24]

---

[23] "Portion Analysis." https://www.youtube.com/watch?v=w1IdFfjRN2I /. Posted 26 Oct. 2018.
[24] "Pasta Meal Test." https://www.youtube.com/watch?v=YXKqbSP0tMw&t= /. Posted 26 Oct. 2018.

Final Testing, Evaluation and Reflection

Stakeholder Re-Visitation

Since I had concluded my previous Initial Evaluation was fundamentally flawed due to the small scope of users reflected on it is therefore not an accurate representation of my wider stakeholders, and not a valid indication my prototype's fitness for purpose. To manage this I sought feedback from a significantly larger number of people from a range of backgrounds.

Testing, Evaluation and Reflection of Prototype with Danijela Unkovich

Due to Danijela already having experience using my prototype she is not an accurate representation of a first time user. Consequently, her responses do not accurately reflect my projects ease of use, an important specification of my brief which I am yet to evaluate on in full. However, I am still able to compare her feedback and experiences from my Initial Prototype with the Final Prototype. This would enable me to determine whether or not the improvements to performance, user interface, and Neural Network accuracy have addressed her suggestions and the broader stakeholders needs identified thus far.

During our discussion when asked whether she felt the app's UI made it easier to interact with the neural network she said "*it helped significantly, being able to point my phone and visually see the food in the camera view makes it much easier to see what the app sees and get nutrition feedback*".

When asked to comment on how well the generated ideal meal portion ratio graphs conveyed an informative picture of 'meal healthiness' she said it "*gave a surprisingly accurate representation of how she herself would rate the meals in real life*". While this indicates an improvement towards fitness for purpose in terms of accuracy, to further reflect on this I ran a performance comparison test between the prototype and Danijela. For a picture of food she and the app would each approximate the meals compliance to the portion ratio and draw a graph accordingly. To also evaluate ingredient inferences both parties would attempt to calculate ingredients. These test results are shown below in Fig-52.

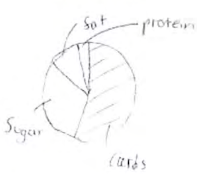
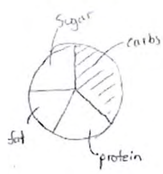
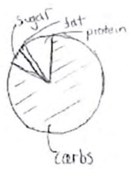
For each meal evaluated, *bread, pasta, soup* and *cake*, the prototype outputted a coherent ingredient summary with measurements, as well as a generated portion ratio graph with a healthiness indication based off Danijelas explored portion ratio. For each test (in Fig-52) the prototype was able to make it's calculation in under 2.4 seconds.

In comparison to its human counterpart Danijela's summaries took, at best, thirteen seconds to complete. Whilst its worth noting, a majority of this time was spent drawing/writing I would expect this to represent the necessary degree of communication to be present when conversing with her clients and in that way it is arguably reasonable to compare her human output to a machine.

Furthermore, my prototype's output closely mirrors, although not 100%, what Danijela wrote. When discussing this with with my stakeholder I noted her liking towards the prototype's delegation of sugar, fat, protein and carbohydrates. During her experience from using my final prototype, she also pointed out ingredient inferences were "*much faster*" from that perceived while using the Initial Evaluation prototype. However, this this is merely one stakeholder's perception I am not able to reliably compare graph generation from my Initial Evaluation to due to it not being fully functional at that point. Despite this, these comments indicate an improvement to overall efficiency in graph generation and prototype performance.

Danijela did not, however, think the colors used in the graph helped convey healthiness with an acceptable degree of clarity. Since parts of the graph they were colored only slightly different shades of green she pointed out this made it somewhat harder to match up its parts to shaded labels underneath. This made me think about people who had colour vision deficiencies and perhaps struggled to differentiate between similar shades of green, undoubtedly raising an important area of future consideration However, it is of course relatively easy to alter these colors to pertain to such requirements so I do not feel that this compromises the fitness of my prototype in light of the aforementioned benefits attributed to the ideal portion ratio inference system.

When asked whether Danijela felt the prototype would function well as a viable tool which could aid in the treatment of her clients she replied "*it definitely shows potential*" but explained the trialing of the app herself for an extended period of time was important before she introduced it into a commercial environment.

Fig: 52	Source:	Prototype versus Danijela - Performance Comparison Test		
Actual Name Inference Duration	White Bread *2449ms*	Pasta *832ms*	Mushroom Soup *463ms*	Cake *451ms*
Meal Image Analysed				
Calculated Name	amish-white-bread	cajun-chicken-pasta	super-delicious-zuppa-toscana	chocolate-eclair-dessert
Calculated Portion Ratio and Healthyness	PORTION RATIO  16% HEALTHY Carbs Sugar Protein Veg	PORTION RATIO  83% HEALTHY Carbs Sugar Protein Veg	PORTION RATIO  33% HEALTHY Carbs Sugar Protein Veg	PORTION RATIO  11% HEALTHY Carbs Sugar Protein Veg
Calculated Ingredients	2 cups warm water (110 degrees F/4 degrees C) 2/3 cup white sugar 1 1/2 tablespoons active dry yeast 1 1/2 teaspoons salt 1/4 cup vegetable oil 6 cups bread flour 39% Confident	4 ounces linguine pasta 2 boneless, skinless chicken breast halves, sliced into thin strips 2 teaspoons Cajun seasoning 2 tablespoons butter 1 green bell pepper, chopped 1/2 red bell pepper, chopped 4 fresh mushrooms, sliced 1 green onion, minced 1 1/2 cups heavy cream 76% Confident	1 pound bulk mild Italian sausage 1 1/4 teaspoons crushed red pepper flakes 4 slices bacon, cut into 1/2 inch piec 1 large onion, diced 1 tablespoon minced garlic 5 (13.75 ounce) cans chicken broth 6 potatoes, thinly sliced 1 cup heavy cream 1/4 bunch fresh spinach, tough stem 33% Confident	1 (7 ounce) jar marshmallow creme 1 1/2 cups white sugar 2/3 cup evaporated milk 1/4 cup butter 1/4 teaspoon salt 2 cups milk chocolate chips 1 cup semisweet chocolate chips 1/2 cup chopped nuts 1 teaspoon vanilla extract 33% Confident
Human Analysis Duaration	*316 seconds*	*390 seconds*	*405 Seconds*	*434 Seconds*
Human Calculated Portion Ratio	 fat protein Sugar carbs	 Sugar carbs fat protein	 protein fat fat carbs	 Sugar fat protein carbs
Human Calculated Ingredients	yeast flour water Salt	pasta cheese Carianda	mushrooms water bread eggs Spring Onions	Sugar eggs flour baking Soda icing Sugar water Chocolate/coca butter

Testing, Evaluation and Reflection with Food Technology Class

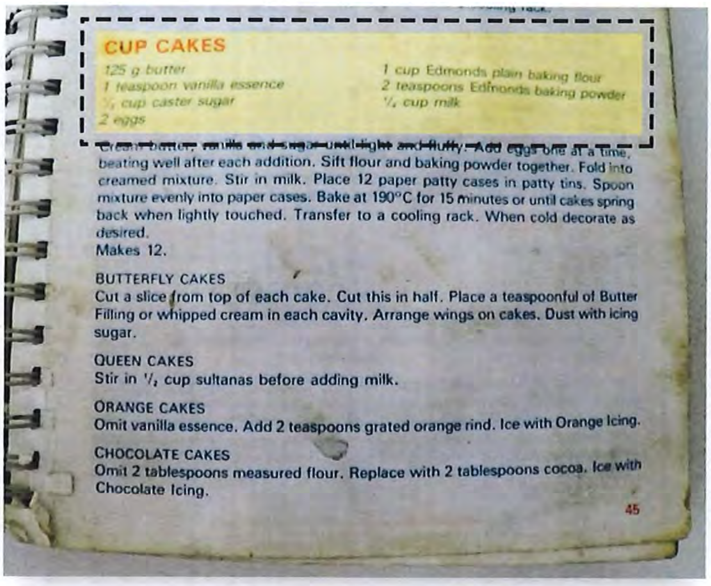
To gain a better sense of how a first time user would react to my prototype I reached out to the food technology class that I had previously worked with. It is important to note although I had held several discussions with this group of stakeholders for research purposes none of them had actual experience using my solution, in either in command line or app form. For this reason I believe they are a valid representation of first time users.

By this stage I had a deployable .apk file of my solution which I distributed amongst class members and this enabled them to easily install my app and test it in a closed environment (this action is evaluated below). I managed to arrange with the teacher for students to use the app during a food technology cooking practical were they would be making cupcakes. I asked each student to use the app to analyse the cupcakes he/she made and then take a screenshot of both the app's generated meal portion ratio graph and ingredient inferences.

To ensure this was a true test of user friendliness I made sure I was not in the room as to discourage students from asking me questions about app functionality. If I have designed to my brief correctly asking questions would be unnecessary as the natural intuition of UI elements should dictate the logical actions for each student to take. I did however ask several students to record their phone's screen during usage so I could later review the actions they took, of which during post analysis, variation in resolution and device make confirmed my solution is indeed cross device compatible. Furthermore, to prove my solution can function independently of an internet connection I instructed all students to switch their device to flight mode.

Fig: 53	Source: self	Real World Cupcake Recipe versus Prototype ingredient comparison

Teacher's Recipe



CUP CAKES

125 g butter
1 teaspoon vanilla essence
½ cup caster sugar
2 eggs
1 cup Edmonds plain baking flour
2 teaspoons Edmonds baking powder
¼ cup milk

Cream butter, vanilla and sugar until light and fluffy. Add eggs one at a time, beating well after each addition. Sift flour and baking powder together. Fold into creamed mixture. Stir in milk. Place 12 paper patty cases in patty tins. Spoon mixture evenly into paper cases. Bake at 190°C for 15 minutes or until cakes spring back when lightly touched. Transfer to a cooling rack. When cold decorate as desired.
Makes 12.

BUTTERFLY CAKES
Cut a slice from top of each cake. Cut this in half. Place a teaspoonful of Butter Filling or whipped cream in each cavity. Arrange wings on cakes. Dust with icing sugar.

QUEEN CAKES
Stir in ½ cup sultanas before adding milk.

ORANGE CAKES
Omit vanilla essence. Add 2 teaspoons grated orange rind. Ice with Orange Icing.

CHOCOLATE CAKES
Omit 2 tablespoons measured flour. Replace with 2 tablespoons cocoa. Ice with Chocolate Icing.

45

Generated Ingredients

```
vanilla cookie cupcake

1 1/2 cups flour
1/4 teaspoon baking soda
1/4 teaspoon sea salt
1 1/3 cups water
1/2 cup sugar
3 eggs
2 teaspoons vanilla
extract
1/2 cup butter
```

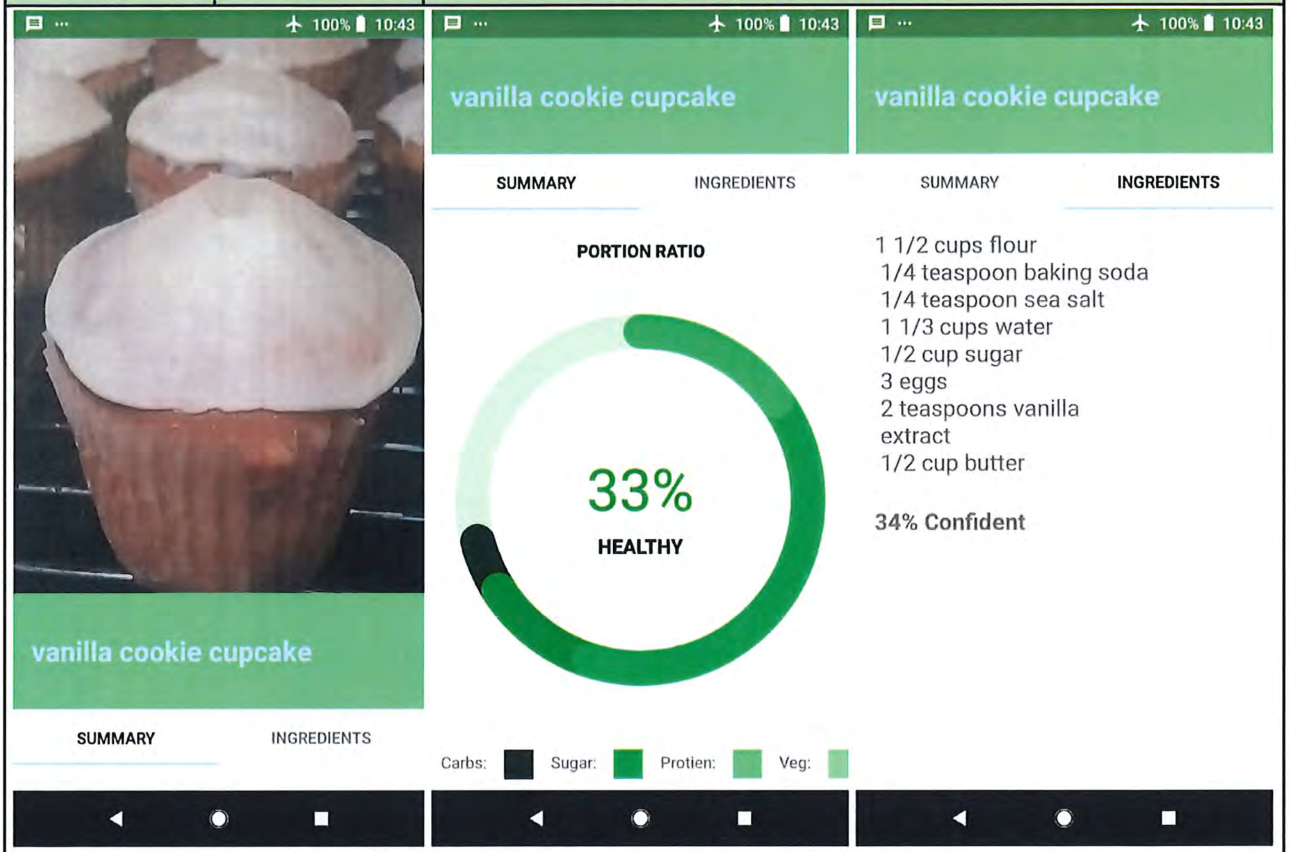
Student Cupcakes

note: For purpose of continuity generated ingredients are shown by recipe convention rather than in metric units

To maintain the ethical nature of my testing practices I ensured all students wrote down whether they prescribed to a specific diet, *ie* veganism. Given the established sensitivity of food in some religions misinforming students to animal products in their cupcakes could seriously offend them. Writing down sensitivities was also an important health and safety consideration given the potentially dangerous effects of students mistakenly consuming allergens such as nuts or dairy. To manage these two complexities I wrote a disclaimer explaining that the app was still in development and every time the app made a classification it displayed a message that data to act as an indicator and not to be considered fully reliable to base medically important and potentially life threatening decision on.

After the class practical one student was apparently able to "*easily and quickly have their cupcake analysed*". I also found this was a common trend during a wider class discussion during a follow up period. Most students commented that the app was successfully able to classify their *cupcake* as a *cupcake* but many reported that their inferred cupcake name was inconsistent with that of their friends, even though they followed the same recipe. For example, the same cupcake was labeled as chocolate in one instance while vanilla in another.

Fig: 54 | Source: self | Screenshots of a classroom test use case while using my prototype



Video of Classroom Test Demonstration[25]

After the practical I was able to compare the actual ingredients from the recipe handed out to students by the teacher between the recorded ingredient output of the app. This comparison is shown above . The yellow highlighted areas indicates meal inferences made by the prototype closely mirror the actual recipe used by the student and suggests a high degree of accuracy is present.

Despite all students using the same ingredients in their cupcakes many were left to customize the toppings, colourings and type of paper wrapping. This led to a myriad of visually different yet deceptively similar cupcakes (a selection of these are shown above in Fig-54) and I feel I can confidently attribute this to the slight variability in the app mislabeling cupcakes as different between students.

However, the overall trend between the main ingredients of the cupcake as 3 eggs, 1 1/2 cups of flower and 1/2 a cup of sugar is mirrored throughout all ingredient inferences and resembles a strong parallel to the teachers recipe (Fig-53). Furthermore, the portion ratio graphs indicating healthiness are relatively consistent with other inferences and true with the teachers own assessment of healthiness.

[25] Youtube Classroom Test Demo https://www.youtube.com/watch?v=w1IdFfjRN2I . Posted 7 Oct. 2018.

For this reason I do not believe this represents a lapse in my prototype's fitness purpose. I also do not feel that since all students had experience in cooking food affected the validity on these findings as I do not feel the skills needed to use the app correlate with one's cooking ability, nor the apps analysis and meal inferences made. However, as a representation of general everyday people I still believe these students accurately reflect the needs of my stakeholders due to this being their first time use. Based on these client re-visitations, I conclude that my project is fit for social purpose at least with regards to fulfilling my stakeholders needs for ease of use and inference adaptability in dynamical problem solving - shown through interpreting the same ingredients in visually different cupcakes.

However, since I distributed the app to students via an .apk file I inadvertently voided the area of reflection pertaining to stakeholders in the real world who would have to navigate either the app store or google play store to locate my prototype and download it. Despite this I believe that interactions within an app store are outside of my control and in that way detriments to the user experience it creates, such as navigational friction, are largely out of my reach and for that reason I don't think it is accurate to evaluate nor penalise my assessment of fitness for purpose.

Fitness for purpose in the broadest sense

Independent Reflection of Stakeholder Needs

Since it is unrealistic for my solution to meet the infinitely variable needs of my stakeholders first outlined at the beginning of this report I can only make a determination of the fitness for purpose if my prototype meets a majority of these needs, reflection so far has shown this is not the case.

While the stakeholder testing conducted above shows usability is largely met by the *first time users* it does not necessarily indicate the prototype's current state is the best approach with regard to their nutrition education needs. This led me to reflect on whether using a graph to convey meal healthiness and compliance with the portion ratio was a mistake. I feel that if implemented successfully, this component outcome would pose a greater benefit for fitness for purpose in terms of both the social and physical context.

For example, while my final prototype does technically meet all requirements laid out within my brief I feel as though the limited dimensionality of feedback a graph can facilitate limits understability somely to stakeholders who are only able to extrapolate its findings to their own meal. This inhibits parallels a graph is able to convey between healthiness and different components of the user's meal.

Conversely, a segmented portion identification and instance highlighting pipeline (Fig-55) would have allowed for potentially the clearest parallel between ingredient inferences and their localisation on a users plate as, by nature, ingredient data would be directly overlaid on the user's analysed picture.

Fig: 55 | Source:https://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/ | Segmented Image Recognition

In reflection I think this would have better facilitated my projects value of usability and gone further in addressing education with regard to the detriment of food choice amongst my stakeholders. However, I still stand by my decision to go with a graph portion ratio as at the time a segmentioned recognition pipeline would not have been technically feasible to implement due to the highly structured and pre-processed dataset that would be required to train my neural network in conjunction with my own personal skill and time constraints. A highly structured and pre-processed dataset also limits the scope of contexts to which I can encompass for my stakeholders in comparison with the dataset needed for nutrition graph generation.

I therefore conclude that the choice of a graph to convey the portion ratio and meal healthiness was the most suitable to implement in my personal context. However, should my project be deemed useful through Danijela in her own commercial use at *Feel Fresh Nutrition* I would recommend a segmented meal analysis pipeline be implemented.

Along with the idea portion ration, another key decision I made during the development of my solution was choosing to build my own custom data set, rather than use a pre existing one, for the purposes of training the neural network. This proved to be a key trade off between breadth and depth but enabled me to structure *image-ingredient* and later *image-measurement-ingredient* example pairs effectively allowing me to better leverage the joint embedding approach to cross model embeddings which Jeff had suggested. In fig 26 this need shaped my solution and formed the basis of an improved general trend in accuracy compared with the initial and final prototype evaluation findings. I, therefore, feel this is a good indication of my prototype's beneficial impact in helping address the issue of nutrition misinformation within my stakeholder's contexts, rather than proving counterproductive through inaccurate feedback as previously thought.

Because of this, I can definitively conclude this decision has been beneficial to the greatest extent possible and allows me to reflect my project meets the values of accuracy discussed Danijela at the beginning of this report. Therefore, from Danijela's perspective, my project meets all foreseeable needs of my stakeholder context and so in this regard is fit for purpose.

Despite this, I have failed to implement functionality within Apple IOS devices as I ran out of time. This prevents me from assessing fitness for purpose within that specific demographic of stakeholders, of which, the effects were partially felt in my classroom test this includes about 30% of the students.

However, given the similarities in user interaction between the two operating systems I do not feel that user experience would be drastically different, had I been able to build an IOS version of my prototype in time. For that reason, I would judge its implications to usability and overall prototype functionality to be relatively similar and not compromise enough to warrant a valid detriment in my fitness for purpose evaluation as a whole.

Technological Trends and Project Values Reflection

It is also important to evaluate my project in terms of its broader role in the key technological trends it is part of, most importantly encompassing the global push towards intelligent systems.

I believe that my project is just a small, but beneficial, step in the right direction. It is physiologically inline with this shift as it proves a functional artificial intelligence to interpret complicated meal information quickly and display it in a stakeholder friendly scannable manner. Machine analysis bears a strong resemblance to how real world nutritionists such as Danijela infer nutritional data from meals.

However, in light of critical reflection my prototype's practice is not fully successful as its implementation of intelligence is yet achieve 100% accuracy. whether such is technically possible is still undiscovered in this field, although in hindsight, my prototype's high (but not perfect accuracy) could in part have been attributed through statically training the neural network, and, failing to utilize self learnability which is a key area of this technological shift. This means that the project can only be used in the present so people using the future 5 - 10 years from now will experience outdated results (this is further explored in life cycle evaluation), and, miss out on what I believe is a core benefit of intelligent systems.

Another key trend that my project is part of is its push for nutrition education in users, moving towards the eradication of health detriments as a result of poor food choice. This has been an important trend and was actually the one I set out to address, I feel the most successful in that regard. The prototype's ability to produce nutritional summaries of equatable, if not greater, quality to those painstakingly generated by humans shown in my testing with Danijela (Fig-51) and recipe comparison (Fig-52), as well as displaying this new information in clear and visually scannable methods allows for outputted nutritional information to be presented in more meaningful way.

Presenting nutrition information effectively was a recurring theme throughout my research, given people suffering from poor food choice today was not brought about not through a lack of nutrition information per se, but through information which is shown in an unreadable and inconvenient manner, often overwhelming the reader and preventing education. By these terms, while limited to the context of nutrition education, my project can also be seen as part of the wider movement toward solving problem of Big Data. A concept by which the amount of available information exceeds the ability for it to be processed and understood efficiently by humans[26]. Based on this, and the considerations for usability made throughout my report, I conclude my project is a highly successful step in facilitating the idea information should have the greatest possible value to the greatest number of people. While rellevent in its own right I would like to use this to emphasise why I think these movements are so important, not only to Dnaiejala's case, by also to a broader society: we cannot anticipate ways in which data will be used to aid in nutrition improvement, proving the power is truly in the hands of individuals to read this data themselves in ways within their own context to affect great social change.

Life Cycle Reflection

An evaluation of my project from a future perspective is important because aspects like maintainability and the changeable nature of technology will impact its fitness for purpose. This also relates to the life cycle of my product, as from past experience, fitness for purpose is never constant and often proves tangible with respect to time.

When distributing the prototype amongst stakeholders in the food tech class Android requires that all apps be digitally signed with a keystore certificate before they can be installed. The validity of this keystore signature can be specified for a period of years by the developer, of which, once elapsed will prevent the app from working correctly. In my solution's case this was set to 5 years. Obviously if the app was to be released and not updated users would eventually find themselves locked out. For security reasons it is impossible to set an unlimited keystore duration, impling especially in Danijela's case, without periodic updates functionality will cease.

In terms of maintainability the Tensorflow Machine Learning Library used in this project is in its early stages of development and so continually changing as it grows and community improvements are made. This may in turn break interactions between my neural network and the app and so the need to continually update could impair the maintainability of my project if new versions are no longer produced. *(Note, tensorflow is licensed under creative comms so it is possible to bundle it's functionality with my project).*

Furthermore, linked in terms of time, perhaps the most relevant life cycle consideration is the neural network itself. Since the final network was base trained to be able to identify 25,314 food types these meals, their ingredients, how people cook and view them will eventually become out of data.

This would completely go against the values of accuracy this project strived for and eventually prove detrimental to its fitness for purpose. This led me to reflect on whether using a static convolutional neural network to interpret meal healthiness was a mistake. I feel that if implemented successfully, this component could only be expected to function correctly for a period of years before data it was trained on, and inherently it's meal inferences thereof, would no longer mirror my stakeholders physical or social

[26] "What is Big Data? | Oracle." https://www.oracle.com/big-data/guide/what-is-big-data.html. Accessed 11 Oct. 2018.

context. While my final prototype does technically meet all requirements laid out within my brief I feel as though its adaptability to the future presents a significant future consideration. Had I considered implementing self training mechanisms to allow the neural network to constantly learn off user data this would have allowed for potentially a more robust solution.

In reflection I think this would have better facilitated my projects value of adaptability to the future and ourselves and gone further in addressing scalability with regard to the detriment of food choice amongst my stakeholders. However, I still stand by my decision to go with a static neural network as the ethical detriments associated with automatic user data collection, as well as data collection itself, would require significant project infrastructure requiring resources and time not realistic to achieve during the development period and not technically feasible in conjunction with my own personal skill constraints.

Therefore, while my prototype's construction and evaluation this far has proved it is fit for purpose and may stay constant, its physical and social environment won't, highlighting the inevitable fact that overall fitness for purpose will change and likely degrate as the app ages. In that way without proper support to maintain its performance it's output may change or completely stop working. So future consideration, should Danijela choose to adopt my solution, I should suggest automatic updates be implemented.

Closing Thoughts

I started Scholarship Technology this year with an open mind. I experimented with several conceptual outcomes before settling on my food analysis machine learning idea. This research allowed me to pursue my stakeholder needs with clarity. I have achieved my aim in making a positive contribution against nutrition misinformation by providing a technological solution for Danijela to give to her clients a better idea of what they were eating.

However, through an ongoing cylindrical development, research and reflection process as well as trialing and experimentation within real world situations I think I have only just scratched the surface within this area of technology.

While in its present state my prototype's ingredient and meal healthiness inferences are key parts, potentially adapting my app to identify allergies in food could go even further to addressing nutrition misinformation or general food safety as a whole. Going further, modifying the same recognition neural network on pictures of people and identifying human health detriments visually, like cancer or obesity, based off CT Scans or MRI images could aid in accelerating medical diagnosis in hospitals.

All in all I believe I have developed a well rounded outcome with a high degree of polish. I have met my brief in full as well as stakeholder requirements, of which, experimentation proved largely successful..