Undergraduate Thesis

Lachlan Perrier

Bachelor of Engineering
Civil Engineering

The University *of* Queensland
2020

Analysis of Transport Preference Survey using Sentiment Analysis

by Lachlan Perrier

Student Number: **45302376**
Course Code: **CIVL4583**
Supervisor: **Mark Hickman**
Submission Date: **03/06/2022**

School of Civil Engineering
The University of Queensland

1. Abstract

A model was created to analyse and predict transport preferences using statistical and machine learning techniques. These techniques correlated the linguistic information with a respondents' behaviour. The analysis was performed on the RACQ MAAS travel survey and focused on predicting Likert scale responses. This survey was not designed with modelling in mind and so the promising results shown indicate that the techniques explored have a promising amount of power.

Various statistical models were created to extend a previously trained language model. The extensions automatically and robustly created logit-like value using only the text prompt and the survey responses. The techniques used to create these statistical models included principal component analysis, auto encoders, and traditional sentiment analysis approaches. The outputs from these techniques were then analysed using logistic regression to predict and understand the survey responses. The models were designed to predict new responses, interpret survey responses, and discriminate between broad heterogenic trends.

Of these techniques the Auto-encoder and traditional sentiment analysis created generalisable logit values of the data that were able to predict the behaviour of the respondents. This performance did not generalise to all questions, but explanatory power was found to be prevalent in some questions.

The all-linear statistical models showed high degrees of interpretability. All techniques attempted to model aspects of the question set related to sentiment (positive and negative). The sentiment analysis-based techniques lead to the most interpretable model which primarily focused on sentiment and with probable secondary considerations on status and definitiveness of a question. These interpretable results indicate that there were broad trends in the data set (such as sentiment) that influenced responses.

Meaningful heterogeneity was uncovered in all models. Statistically significant differences were found between male and females when comparing their respective behaviour measured in the model. The sentiment analysis-based models also observed two separate modes of transport preference behaviour.

These results suggest that language models can be a useful tool when predicting and understanding transport preferences.

2. Acknowledgements

I would first and foremost like to thank my thesis supervisor Mark Hickman for taking on such an unusual project. His knowledge and guidance were a great asset to this project. His experience and ability to provide guidance on a complex multidisciplinary project was indispensable.

I would also like to thank RACQ for proving access to a there MAAS survey data. Without a large and robust data set the methodology outlined would not have been possible. They also provided the survey data in a timely manner, allowing ample time for analysis.

I would like to thank my family for their support during the development of this thesis. Their support and guidance through this process made this report possible.

Lastly, I would like to thank my past lecturers and professors I have had at UQ. The wide breadth of knowledge I have acquired through my past years studying at UQ have made this paper possible.

Table of Contents

1. Abstract 1
2. Acknowledgements 2
3. Introduction 5
   - 3.1 Overview 5
   - 3.2 Aims 6
   - 3.3 Objectives 6
   - 3.4 Scope 6
4. Literature Review 7
   - 4.1 Sentiment analysis and language models 7
   - 4.2 Double Meaning 10
   - 4.3 Negation 10
   - 4.4 Irony 10
   - 4.5 Interoperability 10
   - 4.6 Logit models and utility 12
   - 4.7 Previous work 15
5. Data Set 16
6. Methodology 17
   - 6.1 Choice of Language Model 19
   - 6.2 Dimensionality Reduction 19
   - 6.3 Interpreting Neurons 22
   - 6.4 Logit model Details 23
   - 6.5 Likert scale Conversion 23
   - 6.6 Heterogeneity 24
   - 6.7 Early Stop 25
   - 6.8 Comparison 26
   - 6.9 Comparison Measure (Naïve model) 27
   - 6.10 Naïve Estimation 27
   - 6.11 Exclusion of P-Values 28
   - 6.12 Test/Training Split 28
7. Predictive Power Results 29
   - 7.1 No compression 29
   - 7.2 PCA Compression 31
   - 7.3 PCA Compression 5 Dimension of heterogeneity 33
   - 7.4 Premeasure 35
   - 7.5 Auto encoder 37

7.6 Sparse Auto Encoder 39
8. Heterogeneity Results 42
8.1 No Compression 42
8.2 PCA 5 dimensions 44
8.3 Premeasure 45
8.4 Sparse Auto Encoder 46
9. Interpretability 47
9.1 PCA 47
9.2 Premeasure 48
9.3 Sparse Auto Encoder 49
10. General Evaluation and Discussion 50
10.1 No compression Predictive Power 50
10.2 Auto encoder Predictive Power 50
10.3 Sparse Auto encoder Predictive Power 50
10.4 Predictive Power of PCA Models 51
10.5 Predictive Power of Premeasure Model 51
10.6 Interpretability 52
Sparse Auto encoder 52
Pre-Measure 52
PCA 53
10.1 Heterogeneity 53
10.2 Sensitivity of Analysis 54
10.3 Observations During Model Development 56
10.4 Further work 57
10.5 Future uses 60
11. Bibliography 63
Appendix A – RACQ MAAS Survey 65
Appendix B – Example Code 67
Appendix C – Confusion Matrices 68

3. Introduction

3.1 Overview

Modelling transport behaviour is an important area of research for congestion and business analysis. There are many domains of travel preference prediction, from screen line analysis, traffic count data and survey data. These techniques are generally separable into Revealed Preference (RP), and Stated Preference (SP). The primary concern of this report is SP data, where through surveys (or otherwise), a person states their specific transport preferences after they are prompted.

Typical modelling methodology

Typically, the primary method of analysing SP data involves the use of a logit model. A logit model attempts to model responses by breaking down each prompt into a set of values. For example, a transport option described as “A trip to work that takes 30 min and costs $2.30 in fuel”, could be broken down into the “value of time”, and “value of money” variables. After constructing these values, a person’s travel choice can be determined by analysing a variety of possible trip options and comparing their respective utilities.

There are many advantages of breaking down a survey into a set of attributes which maximise a specific utility:

1) Provides an understanding of how respondents may act in unsurveyed circumstances
2) Provides a single natural measure of “utility”
3) Breaks down reasons a response was chosen

Limitations of this model

This technique of modelling, while effective and clear, has disadvantages. Some of these drawbacks include:

1) Manual definition of values can often be tedious depending on the number of survey questions asked
2) Manual definition of values is subjective when creating a set of questions
3) Surveyor and respondent must agree on what values a specific question is asking, adding a source of error
4) Survey data must be built around a specific set of values
5) Heterogeneity of respondents is measured by blanket inclusion of error terms – not accounting for covariances in survey responses

The primary underlying factor across all these drawbacks is the lack of a natural objective methodology for deriving a set of values from a survey option. Recently, the fields of sentiment analysis and language modelling have begun constructing an objective vector for any given sentence or collection of words.

Commented [J1]: Is this what you meant?

Language Models/Sentiment analysis

The field of data science has recently made great strides in *Natural Language Processing* (NLP). NLP is a field involving the computational analysis and classification of text stimuli. Specifically, a set of techniques have been developed which translate any set of text into vectors, where the vector encodes semantic meaning. In the field of semantic analysis word embeddings have been extremely successful in their ability to extract semantic meaning from sentences.

Further recent advances have been made with the introduction of transformers and entire sentence embeddings. A *sentence embedding* is the computational process of mapping a sequence of characters to a vector (typically high dimensional). These embeddings map any sentence (or small sequence of sentences) into a fixed size vector space. The main feature of these embeddings is similar sentences are mapped to similar vectors. By choosing a suitable distance metric the similarity between two sentences can be measured.

Commented [J2]: In these few paragraphs you have explained what terms mean AFTER you have used them. I have tried to rearrange them so the explanation of the term comes earlier.

By performing the techniques of *sentiment analysis* on these sentence vectors, a set of measures can be performed on a specific survey option. Sentiment analysis is an emerging collection of techniques for deriving meaningful measures of a sentence. Typically, these measures include examples such as “persuasiveness” or “positivity”. The purpose of this thesis is to use sentiment analysis techniques to augment and/or improve the insights survey data on transport behaviour.

3.2 Aims

The primary aim of this thesis is to combine generalisable and interpretable nature of logit models with the statistically objective values extracted from language techniques. This combination of techniques will be applied to transport survey data to provide deeper insights into the nature of peoples’ choices. These techniques will be used to more accurately model respondents’ values and attain a deeper understanding on why people choose the responses they do.

3.3 Objectives

The primary objective for the study is to produce a better model for the respondents’ answers than the equivalent logit model. This model will deconstruct the survey respondent into a set of values upon which to perform logistic regression. The model will predict how a population will respond to unseen survey responses. Heterogeneity of the respondents’ values will be explored in detail, to understand how values change across demographics. The limits of this technique will be further explored and analysed to further understand its abilities.

3.4 Scope

The focus of this report is to investigate a novel method for extracting values from a set of options. The main investigation is focused primarily on feasibility and explores the effectiveness of the proposed techniques. The focus is not on extreme fine tuning of parameters, although some iterative design is included. This analysis is restricted to the transport data provided by RACQ.

4. Literature Review

4.1 Sentiment analysis and language models

Word embeddings are a technique used in machine learning technique for converting a word into a vector. Compared with less sophisticated methods of mapping words to vectors (such as One-Hot or TF-IDF), word embeddings attempt to map to a vector so that interchangeable words are “close together” and different words are “far apart”. Mapping words to vectors in this way allows the given vector to work as the concept of the given word.

Word embeddings are useful in many areas of data science. Some uses include language models, semantic analysis, and document classification. The primary use explored in this thesis is in the domain of sentiment analysis - a method of extracting conceptual meaning from a collection of sentences.

Motivating Examples

It has been found through experimentation that word embeddings produce very similar results to how a person might conceptually map words to vectors. For example, take the table below:

Table 1 - Example Manual Word Embeddings

	Dog	**Cat**	**Puppy**	**Kitten**	**Hat**
Cuteness	7	6.5	9	9.5	1
Aggressiveness	3	4	3	3.5	0
Fashionable	1	0	1	2	9
⋮	⋮	⋮	⋮	⋮	⋮

As can be seen the above mapping would map more words closer together. Using the Euclidian distance norm from on the first three rows $\mathbb{R}^3 \to \mathbb{R}$ we can see:

$$\|f(\text{"dog"}) - f(\text{"cat"})\|_2 = \sqrt{(7-6.5)^2 + (3-4)^2 + (1-0)^2} = 1.5$$

$$\|f(\text{"dog"}) - f(\text{"hat"})\|_2 = \sqrt{(7-1)^2 + (3-0)^2 + (1-9)^2} = 11.22$$

We can see that the above example our intuition, that a dog is more like a cat then a hat, is validated.

Note: Word embeddings can sometimes be sensitive with the specific norm used. Often the cosine similarity is the best measure of distance in word embedding space (Pan, 2020).

Use in Sentiment analysis

Sentiment analysis is an important and effective use for word embeddings. using a set of techniques to measure aspects of a document/corpus of text. These measures could be positivity/negativity, emotiveness, or any adjective to be maximised. Examples in which this method can be deployed include:

1) Social media scraping to understand public sentiment about a specific topic (Themantic, 2022)
2) Analysis of transcripts of congress to track positivity/negativity of politics (Themantic, 2022)
3) Analysis of product reviews (Themantic, 2022)
4) Understanding customer feedback (Themantic, 2022)

Commented [J3]: Is it sensible to quote the same source for all examples? Do you have others?

Sentiment analysis pulls out the aspects of the word vectors which we care about and measures them. This measure can be forced by selecting a specific set of words which best represents the concept. The average is then taken (or some other pooling technique) of these example words. This is the concept by which we measure the remaining words in the data set.

In this paper the typical order of sentiment analysis will be reversed. A logit model analysis will be performed upon a set of measurements of a set of survey prompts. The measurements will be extracted from the high dimensional vector created from the embedding mode that was derived from previous works. The usual techniques of sentiment analysis will then be worked through in reverse to understand these values in a human readable language.

Word2Vec

Word2vec is generally considered an efficient word embedding technique. Published in 2013 by Tomas Manolov and his team, it is a fast and proficient method of constructing embeddings (Tomas Mikolov, 2013). This is specifically important with embeddings as they are often trained on large corpuses of text from a variety of sources to ensure as many connections as possible between words are encoded.

An important aspect of word embeddings is that they are single layered. They encode all their information as a single layer network (Riva, 2021).

Word embeddings are trained on the surrounding context words. Specifically, these models use the surrounding words as the training labels. The Word2vec paper suggests two models; the first uses many surrounding words to predict the current word, and the second uses the current word to predict surrounding words.

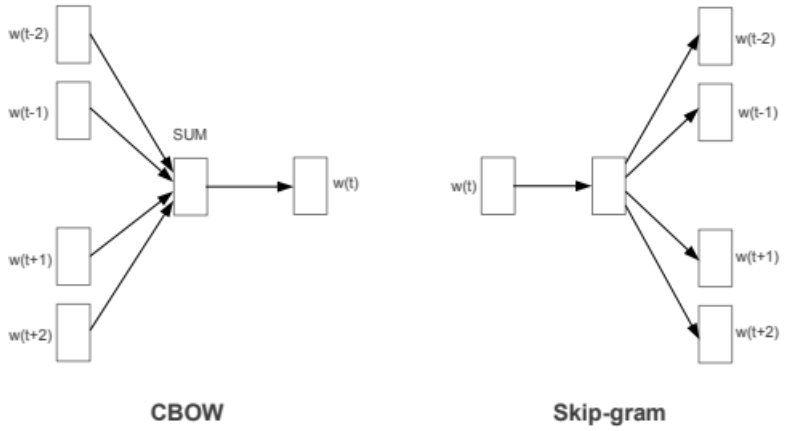


Figure 1 - Overview of Word Embedding models (Tomas Mikolov, 2013)

Continuous Bag of Words (CBOW) takes the surrounding context attempts to predict the current word. These word embedding models can essentially be thought of as small neural networks (Rajamohan, 2018). The embedding space itself is the internal representation of a word in the network. Conversely Skip-gram attempts to predict the surrounding context given a word as an input. In practice both these models are used, as they perform differently depending on how common/specialised a word is in the training set.

Limitations of word embeddings

Word embeddings are generally a good tool when performing low resolution analysis. They are fast to train, and able to represent language in a much more informative way then other techniques like one-hot encoding or strings. However, there are a few limitations that come into effect using the models described above.

4.2 Double Meaning

The first limitation of word embeddings is that there is no room in the model for understanding synonyms. There can only be one encoding for a word, but in the English language, a words meaning changes drastically depending on context (Themantic, 2022).

For example, the word “break” can take on two (or more) meanings (Oxford Languages, 2022).

1) separate or cause to separate into pieces as a result of a blow, shock, or strain.
2) a pause in work or during an activity or event.

The two definitions listed above are contradictory when considering what we have learnt about word embeddings. In a hypothetical sentiment analysis where we are trying to measure the relaxation-inducing qualities of a document, the word “break” doesn’t have a clear mapping. If the first definition is used (like smash or shatter) then “break” is not relaxing at all. However, if the second definition is used then “to have a break” becomes a relaxing concept. To adjust the embeddings based on context, the BERT architecture was invented in 2018 by Jacob Devlin and his colleagues from google. This model adjusts the embeddings based on the surrounding words.

4.3 Negation

Based on what we have seen one might expect the antonym of a word to be the negation of the word.

$$-embedding(\text{happy}) = embedding(sad)$$

Unfortunately, word embeddings don’t work quite like this. Recall that embedding models work by predicting the word from the surrounding context words. This means that antonyms get encoded similarly, since antonyms can often easily replace the original word in a sentence. Care must be taken when trying to correct sentiment analysis models when correcting for negatives. To handle negation more precisely in sentences, the best option is to use a full transformer model.

4.4 Irony

Irony is not understood by word embedding techniques at all. Irony requires a lot of real-world knowledge to interpret. Large language models often struggle to detect ironic comment, making this a difficult domain to analyse.

4.5 Interoperability

Despite techniques used in sentiment analysis, it can still sometimes be difficult to interpret nuanced meaning using word embeddings as words contain many meanings in a variety of contexts. This means that often only broad all-permeating concepts like positivity, negativity or formality can be measured using simple sentiment analysis techniques.

Commented [J4]: Number ten or less are fully typed

Commented [J5]: Is this what you meant? Relaxing-ness is a yucky word for a thesis

Sentence Embeddings (BERT) Overview

Word embeddings struggle when trying to consider the meaning of surrounding words. This limitation means that a model cannot understand synonyms and other more nuanced aspects of language. In 2018 the BERT model (Bidirectional Encoder Representations from Transformers) was proposed to solve these limitations. As stated above, BERT is a transformer-based architecture and will be explored below. BERT Models were invented as an alternative to recurrent neural networks models to process and understand text data.

Commented [J6]: I don't think you have defined this term RNN yet

The primary training of a BERT model is to update the values a word-embedding depending on its context. This means that a BERT model could update the embedding of a word with many synonyms to the one most appropriate to a specific context. BERT models also handle negation more efficiently, as the context allows the meaning to be updated by the surrounding words.

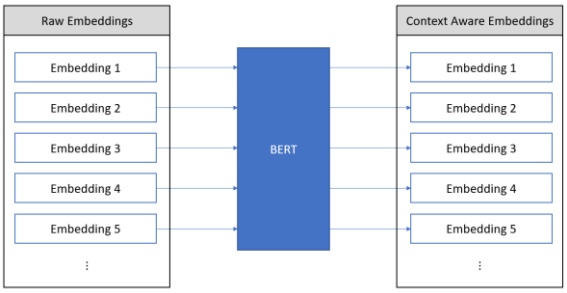


Figure 2 - Bert Architecture Overview

Sentence Transformers

Note that BERT doesn't create a single representation for a sentence. The context-aware embeddings are used primarily in translation and text prediction models. However, BERT models have also found to be useful in the field of sentiment analysis, which require a single combined vector. To transfer a corpus of text into a single vector, a variety of transfer learning tasks are applied to achieve the desired fit. These extra tasks transfer what the model has learned about language to create a single sentence embedding for a piece of text. This general methodology was created in the seminal paper *Attention is All you need* (Vaswani, 2017).

More sentence embeddings have since been built upon the original BERT architecture. Most of the improvements have primarily focused on creating bigger models with more data to provide a better understanding of language. Some of these models include:

- XLnet
- RoBERTa
- GPT-3 Embedding Models

These BERT extension models will not be explored extensively as much of their meaning is lost in down sampling for use in logit models.

4.6 Logit models and utility

Choice modelling is the process of analysing choice data to predict choices. Choice models typically break down a of options into their respective utility. Utility is the numerical representation for preferability of a given choice. Logit models are the primary methodology of breaking down and comparing different aspects of a respondent's answers into utility.

Choice theory

Choice theory is mathematical analysis of the decision process of certain choices. Specifically given a set of choices C

$$C = \{C_1, C_2, C_3 \dots\}$$

A function (f) can be defined that map each of these choices to a utility:

$$f: C \to \mathbb{R}$$

Where the utility is the relative value of a given choice. Given the utility each element of a set of choices, probability of any individual choice being chosen can be defined as (Savage, 2019):

$$P(C_m|C) = \frac{e^{f(C_m)}}{\sum_c e^{f(c_i)}}$$

This formula follows from assuming that there is an error term on the utility:

$$U_m = f(C_m) + \epsilon_m$$

Where $\epsilon_m \sim EV(0, \mu)$

Then:

$$P(U_m \; is \; greatest \; utility) = P(f(C_m) + \epsilon_m \geq \max_{n \neq m}\big(f(C_n + \epsilon_n)\big)$$

This methodology means that the primary design decisions when modelling any set of decision is the construction of the utility function f.

Choice Data:

Choice data can generally be broken down into two categories. These categories are Stated Preference (SP) and Revealed Preference (RP). Revealed Preference is characterised by peoples' actions in a real-world decision. In transport these include traffic counts, car sales and public transport data. While this data is generally preferable it can often be difficult to find for a specific set of questions. Stated Preference is usually derived from survey data. It is useful due to its versatility and ability to prompt a sample on any set of questions. This thesis' main domain is the analysis of Stated Preference data.

In some data sets, socioeconomic attributes can be included in the utility model. An example of this might be the value of time of a high socioeconomic status person, is much greater than that of a low socioeconomic status person. These attributes can be modelled by further implementations to the utility function.

Utility functions:
Utility functions are usually manually constructed using knowledge engineering. Specifically utility function can be constructed as a set of values. An example for a given choice would be:

$$C_1 = \begin{Bmatrix} travel\ time = 15\ minutes \\ cost\ of\ travel = \$3 \\ environmentally\ friendly = 1 \end{Bmatrix}$$

Then:

$$f(C_1) = \beta_1 \times 15 + \beta_2 \times 3 + \beta_3 \times 1$$

Where:

β_1: value of time

β_2: value of money

β_2: value of environmentally friendly option.

The attributes of a utility function can be

- Generic (value of measure is universal)
- Specific (value of measure is specific to travel option)
- Quantitative (e.g. cost, travel time)
- Qualitative (e.g. environmentally friendly)

The primary purpose of this thesis is to construct a methodology of building these utility functions out of sentence data that does not have an obvious basis of measurements.

Extensions to logit model:

Modelling heterogeneity:

Heterogeneity is typically modelled by the addition of error terms to the utility function. However, this makes calculations of the probabilities more difficult due to these extra error terms. For example, if an extra error term ϵ_m is conditionally added to two of three distributions

$$\int_{\epsilon_m} \frac{e^{U_1+\epsilon_m}}{e^{U_1+\epsilon_m} + e^{U_2+\epsilon_m} + e^{U_3}} f(\epsilon_m) d\epsilon_m$$

we see that this equation has no analytical solution. This poses some problems in the methodology section of this paper, and thus a different method of heterogeneity is constructed.

Ordered logit models:

Ordered logit models are used when the result has a categorical order. Specifically, the utility is mapped to a corresponding set of possibilities. Given a utility U, the choice $y \in \mathbb{N}$ can be modelled as:

$$y = \begin{Bmatrix} 0 & if\ U \leq \mu_1 \\ 1 & if\ \mu_1 < U \leq \mu_2 \\ 2 & if\ \mu_2 < U \leq \mu_3 \\ & \vdots \end{Bmatrix}$$

Using the assumption of $U \sim EV$ the probability of each value of y can be calculated as:

$$P(y = n) = \frac{\exp(U - \mu_{n+1})}{1 + \exp(U - \mu_{n+1})}$$

This allows for the prediction of the probability of a given choice.

Commented [J7]: Allows what?

Regret models:

Regret based models primarily focus on the tendency of people to make choice based on what would minimise regret. This loss aversion leads to different behaviour than pure utility maximisation.

Nested Logit:

Nested logit models account for different attributes containing similar values. An example would be if two preference options involved a blue bus or a red bus. The important aspect of these models is the bus attribute, and separately the colour of the bus (Hausman, Econometrica).

Other:
The field of logistic models have many other possible combinations. Generally, all the extensions to the logit models revolve around relaxing assumptions of the original logit model. The primary nature of this work is to assess the utility of using language models to augment inputs to a logit model, so overly complicated combinations of logistic models will not be implemented in this paper.

Traditional Modelling Process

Traditionally a set of questions is constructed around the logit model. For example, if a study is constructed to explicitly model value of time against value of an environmentally sustainable option. A survey might question might be constructed as:

"I would use a more environmentally friendly transport if it took an extra 5 minute"

- Agree
- Disagree

By changing the changing the time and recording the responses the options can be fed into a logit model:

$$\beta_{env} x_{env} + \beta_{time} x_{time} = U$$

Since there are no alternatives the utility function simplifies to:

$$P(\text{Agree}) = \frac{1}{1 + e^{-U}}$$

Where $\beta_{env}, \beta_{time}$ are fit using logistic regression and (for the above prompt) $x_{env} = 1$ and $x_{time} = 5$.

The purpose of this thesis is to find an automated and objective process of determining the set of x values. This will give a better representation of the factors that give utility.

Commented [J8]: Remove page break

4.7 Previous work

The principal study on which this work is based is *Predicting Survey Responses: How and Why Semantics Shape Survey Statistics on Organizational Behaviour* (Jan Ketil Arnulf, 2014). This study explored the use of Latent Semantic analysis (LSA) for embedding survey data. LSA is a technique of semantic analysis which relies upon grouping documents by grouping together correlated words. Specifically, LSA takes the counts of each word in a document and encodes it as a vector. For example:

$$"The\ cat\ is\ big. The\ cat\ is\ scary"$$

Is mapped to:

Table 2 - LSA Data Example

Word	The	Cat	is	big	scary
Count	2	2	2	1	1

A singular value decomposition is applied to the collection of documents. Namely for a matrix constructed as the concatenation of document vectors X

$$X = USV^T$$

where S has all its values on the leading diagonal and U, S are orthonormal matrices. This gives a set of measurements of each document to analyse.

Arnulf's paper used this technique to map a set of documents into a set of vectors. The cosine similarity was then used to assess a question's similarity to the other questions. This similarity measure was then used to group a question to its closest counterpart, and to subsequently predict survey response data.

Arnulf's paper contains a second method that individually compares the similarity of words. This specific methodology is less related to the current work and makes heavy use of knowledge engineering to account for various defeciencies.

Arnulf's paper successfully applied this technique to four data sets. The prediction accuracy ranged from 60–86% based on a respondent's surrounding values. This field is relatively new so there are few other works specifically relating to survey response prediction.

This paper was not used as a baseline in this study as the text embedding techniques are primarily focused on grouping the prompt documents instead of interpretation. The data set used for analylis in this report has the questions human curated into various catagories as outlined in section four. Instead an analysis will be performed on individual subsections of the questionnaire.

5. Data Set

The data used in this study is the RACQ MaaS survey conducted in 2019, in which941 participants responded to the survey results. The questions were presented in the following format:

Q16.2

It is easy for me to travel to and from public transport (e.g. stops/stations are close by or have parking or good access via footpaths/bike paths).

1	Strongly disagree
2	Disagree
3	Somewhat disagree
4	Somewhat agree
5	Agree
6	Strongly agree

The survey has seven main categories in which respondents are asked their opinion:

1) Public Transport and Active Transport
2) Alternative Transport
3) Journey Planning and Paying for Travel
4) Car Subscriptions
5) Mobility as a Service
6) Electric Vehicles
7) Automated Vehicles

Models will be constructed on the entire data set. The full survey and all prompts are presented in Appendix A.

6. Methodology

Overview

The model will use different methods of sentiment analysis to deconstruct transport preferences. The proposed technique uses a language model to automatically construct a logit model from the stated preference data set. This paper relies heavily upon previously constructed sentence embedding models to transform all prompts into a statistically interpretable vector space.

The proposed model can be broken down into 3 primary steps:

1) **Language model:** Using a previously trained sentence embedding model such as BERT or XL net. This step takes a string of characters as an input and outputs a high dimensional vector representing the sentence.
2) **Dimensionality Reduction:** Many techniques will be applied to the high dimensional output to reduce it to a set of "measurements" of a scenario. A logit model cannot be directly applied to the raw language model output due to overfitting concerns. This process has many techniques which are described below.
3) **Logit model:** A logit model will be constructed on the output measurements to predict the survey response. This model will be constructed using the Pytorch library for flexibility and automation.

Commented [J9]: What is this?

The entire process is shown in Figure 3 - Model Overview.

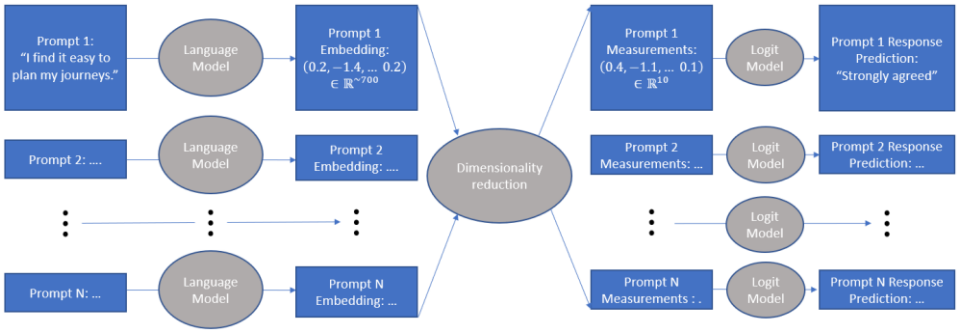


Figure 3 - Model Overview

Any set of sentences can technically be modelled using this technique. Some dimensionality reduction techniques (such as PCA) require the entire set of embeddings to construct principal directions. However, these dimensionality reduction techniques are still to be generalisable to unseen sentences.

Output

The primary benefit of this methodology is to find a more fundamental basis by which decisions are made. This fundamental basis will provide more accurate survey prediction, more intelligent heterogeneity modelling and more meaningful measurement of a survey respondents values.

Gradient descent
The proposed methodology revolves heavily around the usage of gradient descent and backwards propagation to fit the model parameters. Although some of the sub models in this report could be solved with other techniques and methodologies, gradient descent was preferred for its relative flexibility and its ability to be used as a consistent technique as various other aspects are changed in this study. Gradient descent was used for all models to allow consistency.

Loss functions are used in regression when finding a line of best fit by minimizing the overall inaccuracies of the model. The loss function used in this study is the Binary Cross Entropy Loss (BCE) as suggested by Godoy in his article (Godoy, 2018).

Technology stack
The methodology outlined below is implemented within an Anaconda Python environment and run on a Jupiter notebook. The following packages were included in the python environment.

- Pytorch: This methodology required the fitting of the arbitrary mathematical models. Specifically, the primary features of the Pytorch library including gradient descent
- SentenceTransformer: package used for wrapping and handling the BERT language model.
- Seaborn/Matplotlib: used for data visualisation and result validation
- Scipy/Sklearn: Python modules used for additional statistical and data processing

This technology stack was chosen due to its relative flexibility. This allows for low level control of the entire model pipeline. Using gradient decent, all parameters of the model are fitted to any set of data. Performing an empirical best-fit in conjunction with the usage of language models allows for a complex and nuanced set of models to be produced.

Benefits of Proposed methodology
The main benefit that this model offers over a traditional logit model is that it automatically constructs a set of values that the user is trying to optimise. This will give deeper insights into the reasons why people make the decisions they do. An example of could be the sentence:

"An environmentally friendly and expensive transport option"

Commented [J10]: Did you mean Inexpensive?

which in a traditional logit model would be mapped to a vector made by an engineer such as:

Table 3 - Engineered Logit Model Values

Value of environment	Value of money	Value of time	Value of Privacy
1	1	0	0

However, the internal process of the respondent might have made their decision using a completely basis for their decisions:

Table 4 - Respondent Values

Value of Status	Value of quietness	Value of time	Value of Convenance
1	-0.5	0	0.5

It is important to note that these two vectors may not necessarily be a linear transform of each other. If they are linearly dependent, then the solution would arise in the covariances of a traditional logit model.

The purpose of this methodology is intelligently constructing this vector using machine learning and language models. By constructing these vectors, this model will be more accurate and have a wider domain of predictions.

6.1 Choice of Language Model

In the field of natural language processing, there exists many different sentence embedding models. The language model for this paper is the 2018 BERT model. This model is free to use and constructs excellent embeddings, extracting large parts of sentence similarity. Other language models exist including XLnet, GPT-3 and RoBERTa. It is expected that the overall results will not be sensitive to original embedding model. The dimensionality reduction techniques mean that a high-quality model (which typically outputs higher dimension embeddings) will only output superfluous extra values.

6.2 Dimensionality Reduction

Dimensionality reduction is one of the key engineering aspects of the presented models. Dimensionality reduction attempts to map a sentence embedding, to a set of measurements or values that a sentence is compared against. The dimensionality reduction techniques are processed with logistic regression to attain utility. This utility can be used to predict respondents' behaviours.

Principal Component Analysis

Principal Component Analysis (PCA) is a commonly used dimensionality reduction technique on high dimensionality data sets. For a given data set M a PCA solves the equation

$$M = UDV^T$$

where U and V columns span an orthonormal basis for M, and D is a diagonal matrix with each diagonal value corresponding to the variance of the data along the U and V dimensions. In this methodology PCA is applied to the set of embedded sentences. Given the function from a sentence to a vector $E(s_i) = v_i \in \mathbb{R}^{\sim 1000}$, the sentence matrix is constructed as so:

Given a set of sentences

$$\begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{bmatrix}$$

We can construct a sentence space:

$$\begin{bmatrix} E(s_1) \\ E(s_2) \\ \vdots \\ E(s_n) \end{bmatrix} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = UDV^T$$

The set of values which will be fed into the logit model can simply be taken as: $values = UD$

Auto-Encoder approach

The autoencoder approach will constructed a deeper neural network to create a set of values with which logistic regression performed. An auto encoder of the text embedding space will be constructed as outlined in the paper of Stewart. (Stewart, 2019), such that

Encoding: $z = \sigma(Wx + b)$

Decoding: $x' = \sigma(W'z + b')$

where the logistic regression is applied to the latent space z. This encoder is made sparse by implementing a sparsity constraint on the encoding matrix.

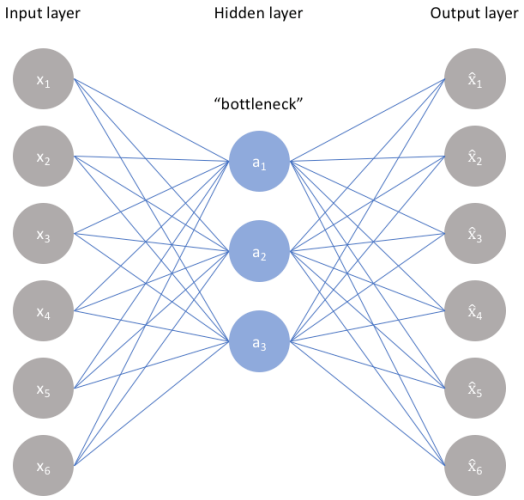


Figure 4 - Auto Encoder Architecture (JORDAN, 2018)

Sparse Auto encoder

The sparse auto encoder was an auto encoder model with an additional constraint added to the data to reduce overfitting concerns. An additional sparsity constraint was added to the loss function which penalised the auto encoder for larger weights. This led to the final equation for loss as:

$$loss = BCE(pred, truth) + \lambda \times \sum_{i} (W_i)^2$$

This reduces overfitting concerns and leads to a better performing model.

Premeasure

The closest technique to traditional sentiment analysis involves the pre-measure technique. The premeasure technique inputs a list of adjectives into the language model (Hugsy, 2022). The cosine distance (defined as $\|A\|\|B\| \cos\theta$ with theta being the angle between A and B) is used for calculating the similarity of a sentence prompt to an adjective. The similarity of each option to the adjectives is than fed into the logit sub model for predicting utility.

The comparatively large usage if say 10 000 adjectives are used allows for the methodology to have the high explanatory power. A more interpretable model could possibly be created by limiting the adjectives used for comparison. Limiting the adjectives would likely lead to a less powerful model. The restriction of the input words to adjectives is primarily focused around describing the higher order concepts of each chosen option. Other word types, such as nouns, were ignored as they would not focus on the essence of the text.

Commented [J11]: Check this sentence. Is this what you meant?

Comparison of all Models

Commented [J12]: I don't understand where you for the number of parameters from.

The models outlined above are listed in table 5. The number of parameters are the number of free variables that are fitted using backwards propagation. A high number of parameters implies a high chance of overfitting.

Table 5 - Comparison of Compression Techniques

Compression technique	Notes
None	- Linear model - Non interpretable - Many parameters ~ 4164 (high chance of overfitting)
PCA	- Restricted linear model (subset of None) - Interpretable - Small set of parameters ~1980 (low chance of overfitting)
Auto Encoder	- Nonlinear model - Interpretable - Many parameters ~ 9580 (very high chance of overfitting)
Sparse Auto Encoder	- Nonlinear model - Interpretable - Many parameters ~ 9580 (lower chance of overfitting)
Premeasure	- Nonlinear model - Interpretable - Small set of parameters ~ 2324 (low chance of overfitting)

6.3 Interpreting models

One of the primary objectives of the model architecture is to produce interpretable results. The interpretability is attained by observing the outputs of the compression method. These outputs are the physical measurements that will be multiplied by the corresponding beta values to calculate utility.

To interpret these typically abstract values outputted from the compression technique, a list of approximately 10 000 adjectives was fed into the trained model. The adjectives that resulted in high beta coefficients and had a high effect on utility were analysed. These adjectives were sorted from highest activation to lowest activation.

The highest activation results were reported in Section Eight. Note that this methodology is specified to broad relationships and is not as effective at detailed analysis. Generally, sentiment analysis techniques use a similar methodology of comparing the embeddings of a corpus of text to a set of adjectives.

6.4 Logit model Details

A logit model was constructed using the Pytoch library and solved using gradient descent. Specifically given any set of values V the model will be constructed as

$$U = V \times W^T + \beta$$

where W^T β are the parameters of the model, and U is the output utility. The probability of the option being chosen is then calculated in the usual multinomial logit methodology. This construction is necessary to allow back propagation for dimensionality reduction techniques.

6.5 Likert scale Conversion

A specific difficulty experienced in the methodology of this report is the encoding of the Likert scale. Traditional logit models solve equations with Monte Carlo methods, which do not allow a gradient to backpropagate.

Commented [J13]: Do you mean Monte?

In a typical machine learning classification task, categorical encoding is achieved using OneHot encoding.

Table 6 - Example One Hot Encoding

Number	OneHot
1	$[1, 0, 0, \dots]$
2	$[0, 1, 0, \dots]$
3	$[0, 0, 1, \dots]$

However, Likert scale data used in this report is ordered. This gives more information as to how the options should be modelled.

The method is outlined in this Grubers paper (Gruber, 2021). This method encodes an ordered logit model as a binary classification task where each logit option is encoded as shown in Table 8 - Encoding of ordered logit.

Table 7 - Encoding of Ordered Logit

Response	Number	Encoding Tensor
Strongly Disagree	0	[0, 0, 0, 0, 0]
Disagree	1	[1, 0, 0, 0, 0]
Somewhat Disagree	2	[1, 1, 0, 0, 0]
Somewhat Agree	3	[1, 1, 1, 0, 0]
Agree	4	[1, 1, 1, 1, 0]
Strongly Agree	5	[1, 1, 1, 1, 1]

Using the predicted utility calculated as outlined above (U), the utility vector is calculated as using the Likert function L

$$L(U) = \sigma(U[1, 1, 1, 1, 1] \cdot [c_1, c_2, c_3, c_4, c_5])$$

or equivalently

$$L(U) = [\sigma(U + c_1), \sigma(U + c_2), \sigma(U + c_3), \sigma(U + c_4), \sigma(U + c_5), \sigma(U + c_6)]$$

where σ is the sigmoid function:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Due to the above encoding of the ordered logit, after training it must be:

$$c_1 \geq c_2 \geq c_3 \geq c_4 \geq c_5$$

since any other order would lead to a worse performing model. This technique allows for traditional utility to be calculated as normal.

6.6 Heterogeneity

A novel approach to modelling heterogeneity is used in this investigation. Traditional methodologies require computational methods to solve the final logit model. This is computationally expensive and does not provide back-propagation, which is essential for the proposed modelling methodology. To allow for backwards propagation through the model, heterogeneity will be modelled as a function of a OneHot representation of survey responders. OneHot encoding is a commonly used representation in machine learning models where the n^{th} respondent is represented as a zero vector with a 1 in the n^{th} place. This function is expressed as:

$$p_1 = [1,0,0,0 \dots n\ times \dots\ 0]$$

$$p_2 = [0,1,00 \dots n\ times \dots 0]$$

then

$$W_n = W_0 + p_n AB$$

A: $n \times a \rightarrow$ number of survey responders × number of dimensions by which respondents vary

B: $a \times w \rightarrow$ number of dimensions by which respondents vary × weight deltas of a respondent

where W_0 are the global weights for each person, n is the number of survey respondents, A represents the variability by which respondents differ and B translates these dimensions to deltas to be applied to the weight matrix. The above is formally known as an outer product decomposition and is commonly used in machine learning to constrain a low rank matrix (Pal, 2018).

Demographic analysis can then be performed using simple regression techniques against each dimension of variability expressed as:

$$pA$$

which if a is set to 2 in the model (as it commonly is), the first person's heterogeneity parameters would calculate as follows:

$$pA = [1, 0 \dots] \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ \vdots & \vdots \end{bmatrix} = [a_{11} \quad a_{12}]$$

These heterogeneity parameters are then converted into vectors of betas with the B matrix

$$[a_{11} \quad a_{12}] \begin{bmatrix} b_{11} & b_{12} & \dots \\ b_{21} & b_{22} & \dots \end{bmatrix} = [\beta_1 \quad \beta_2 \quad \dots]$$

where v_n is the values or the correlative coefficients between a certain feature extracted from the language model, and the measure utility of a chosen response. This leaves the final utility model for each person as:

$$U = \beta_1 c_1 + \beta_2 c_2 \ldots$$

where c_n are the meaning derived from the language model, and β_n are the relative beta coefficients for each respondent.

6.7 Early Stop

A common difficulty when fitting arbitrary models to a given latent space is that gradient descent-based techniques don't necessarily fit the data optimally. When there is a limited amount of data available, overfitting is a concern. Overfitting is an issue found in many complex models, where the model fits the noise found in the data set rather than the broad overarching trends in the data itself. A visual example of this can be seen in Figure 5.

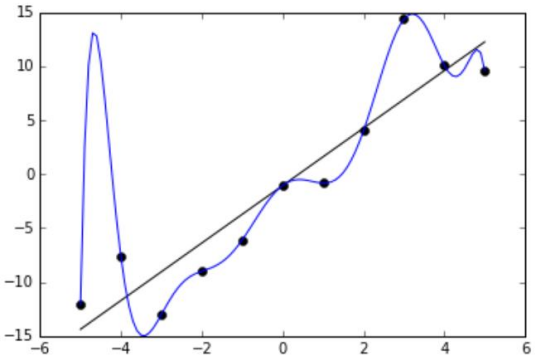


Figure 5 - Example of Overfitting

Ideally this is solved by building a model that is small enough to not overfit. Some models in this report are small enough to not overfit, but many are not. To limit the problems caused by overfitting the models' training was cut off at an early point in the process. This stopped the model from fitting noise found in the data as well as well as giving more generalisable results. In this study a test training split was created, and the models were trained on the training data until no more improvements were obtained in the test data.

6.8 Comparison

A unique difficulty in this project is creating the appropriate metrics against which to measure the quality of the results. The primary and objective measure of performance of a model is its ability to predict survey results from unseen test questions. This fundamentally tests the aggregate model's generalisation of the problem. Due to the questions set not being designed around this methodology, some test questions will perform poorly. In the following results, a particularly high-performing question (Q22.2) was separated from the test set to observe an "upper bound" on the predictability.

Difference squared

The "goodness of fit" measurement that was decided upon was difference squared.

$$difference\ squared = \frac{(true\ answer - predicted\ answer)^2}{number\ of\ answers}$$

Where each answer is encoded as an integer encoded as shown in Table 8 - Encoding of Survey Responses.

Table 8 - Encoding of Survey Responses

Answer	Encoding
Strongly Disagree	0
Disagree	1
Somewhat Disagree	2
Somewhat Agree	3
Agree	4
Strongly Agree	5

The difference squared measure is primarily focused on the accuracy of the model. A small difference squared would imply a model with high confidence in its predictions.

R^2

The second measure of the quality of the model is the Pearson R squared. This measure informs the amount of variability that the model explains. A high R^2 indicates that the model has a high discriminatory power when predicting the survey responses, even if the prediction of the response is incorrect. A model that has a high R square but a low sum of squared implies that the model outputs may be able to be "corrected" with further sample data. The formula for Pearson R squared is taken as below:

$$R^2 = \frac{\sum(true\ answer - predicted\ answer)^2}{\sum(true\ answer - average\ answer)^2}$$

6.9 Comparison Measure (Naïve model)

Ideally the predictions from the models would have a robust previous methodology to compare its performance against. A robust logit model comparing a set of the respondents' values could potentially serve this purpose. Unfortunately, the specific data set used in this study was not designed with logit modelling in mind. As a result, any logit model will lack generalisation ability to external questions.

An example on a small subset of questions might be Q22.3, Q22.4 and Q22.4

Code	Question
Q22.2	I like the idea of safe automated vehicles if they make it easier for me to travel.
Q22.3	I trust automated vehicles to operate safely.
Q22.4	I would consider buying an automated vehicle.

This subset could result in the following measures for a logit model.

Code	Value of convenience	Value of safety	Value of money
Q22.2	1	0	0
Q22.3	0	1	0
Q22.4	0	0	1

This logit model would simply run the mean answer for each question without providing much insight into how a set of respondents would respond to a novel question. Instead, a naive estimate was used as a baseline to compare the model with.

6.10 Naïve Estimation

Instead of producing a logit-based model, a naïve estimation will be created where the average of each individual respondents' responses is taken. This measure attempts to account for the general enthusiasm of the respondents and how generally agreeable they are.

In the naïve model, if a respondent only responded to three questions as "strongly agree", "Agree" and "disagree" then their predicted response (using values from Table 8 - Encoding of Survey Responses) gives a single respondents predicted answer as:

$$\frac{5+4+1}{3} = 3.33 \rightarrow somewhat\ agree$$

If the language-based model performs better than the naive base, then the model must understand the fundamental patterns in the text that lead to the response. In the following results, the model predictions are compared against the naïve estimation to evaluate their efficacy.

The rational for this methodology follows from formulating the naive base as a logit model. This model would be the "most powerful" whilst taking no information from the sentences. Since – as mentioned above – it is difficult to observe meaningful beta coefficients, the only explanatory power could come from the heterogeneity parameters.

Commented [J14]: Can you phrase this differently

$$U = \beta_{male} I_{male} + \beta_{female} I_{female} + \beta_{income} I_{income} + \cdots$$

Where I are indicator variables and β are the respective coefficients

The asymptotic limit of this modelling approach as more parameters are added approaches:

$$U = \beta_{respondent\ 1} I_{respondent\ 1} + \beta_{respondent\ 2} I_{respondent\ 2} + \cdots$$

where each respondent has its own beta values. These beta values denote their general agreeableness with respect to a general prompt. This methodology does not attempt to interpret the question but gives the most explanatory power possible without using textual data.

Commented [J15]: Doesn't make sense

6.11 Exclusion of P-Values

Regular logit models avoid "overfitting" by creating a set of p values for each regression coefficient (β). This allows a further analysis by removing the insignificant coefficients from the model. This eventually leads to a robust model that only models significant effects. The Pytorch-based technology stack that is required for full backwards propagation does not return a set of p values for each parameter it estimates.

With the neural network-based approaches undertaken in this report, it is atypical to remove insignificant coefficients. Instead, a training/test split is used to evaluate how well the model generalises, and how spurious the relationships found in the model are. This is standard practice when constructing large models in a data science context.

P values are used in this report when it is appropriate, IE when comparing heterogeneity parameters, the means and variance of two sets of data are compared for statistical significance.

Commented [J16]: Doesn't make sense

6.12 Test/Training Split

As is standard in data science, the survey questions were separated into "training" and "test" data sets. The training dataset is used to estimate a model's parameters, including heterogeneity and base values. The model is validated by comparing its predictions of the test set using the model created from the training set.

The study was conducted on a single set of Likert scale questions. Three test questions were removed including Q20.2, Q21.2 and Q22.2 from the training set. These questions were randomly selected before analysis for comparison. It is important to note that one, two or all three of these questions may fail to predict effectively. Ideally the model would generalise to these questions, but a failure to generalise is not catastrophic. If the model cannot predict the answer to an unseen test question, this may be because there is little correlation between the test set and the unseen question.

7. Predictive Power Results

7.1 No compression

The first model created was set of logit weights where no compression was used. This model simply fits a logit model to the raw embeddings produced by the BERT model. This Specific model involved 2 dimensions of heterogeneity.

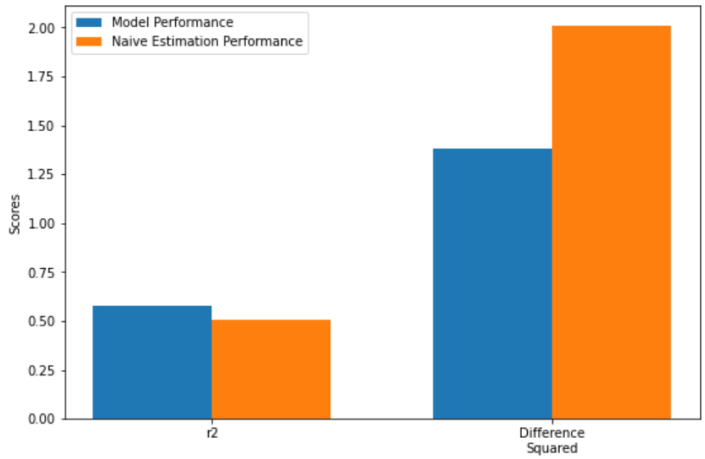


Figure 6 - Model Performance on Training Data

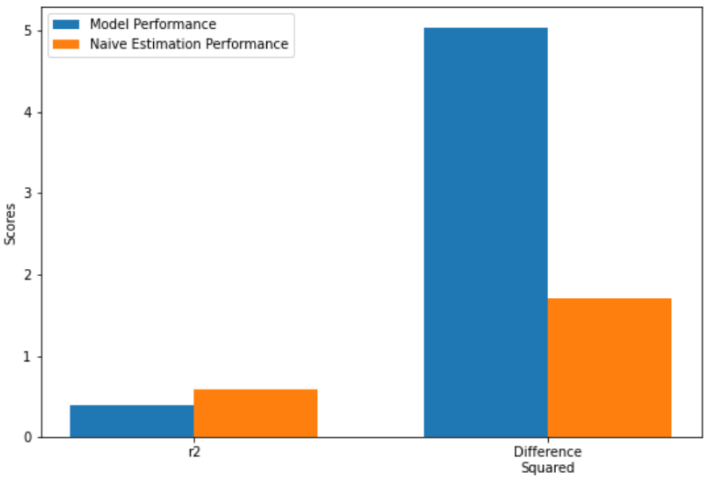


Figure 7 - Model Performance on Test Data

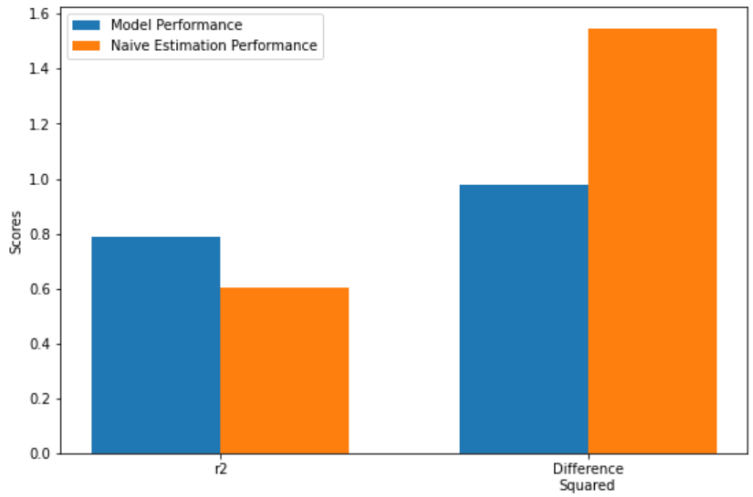


Figure 8 - Model Performance on Q22.2

Despite the relative simplicity and excess of parameters, the “no compression” methodology was still able to fit the training data. The entire model was able to improve on the naive control from 0.50 to 0.56. The no compression model also maintained a better difference squared from 2.1 to 1.32.

The model did not generalise to all test questions, performing significantly worse than the naïve model. This was overall expected as the diverse questions in the data set were unlikely to be effectively modelled wholistically. As can be seen in figure 7 the models attempt to predict the test questions lead to a weak prediction, with little explanatory power.

Question 22.2 was the best performing of all the test questions. In this question, the model attained an accurate prediction of the responses. This strong generalisation ability indicates that the model has developed a semantic understanding of the question in a general manner.

These results are a reasonable “baseline” for the generalisability and accuracy potential of this model. The ability to predict some – but not all – out of domain questions indicate that the model has some fundamental understanding of the decision process behind the Likert scale responses. An important note is that this model is a superset of all linear models, as any linear intermediate compression layers will lose information compared to this model. The primary advantage that the other models will maintain involve generalisability and interoperability.

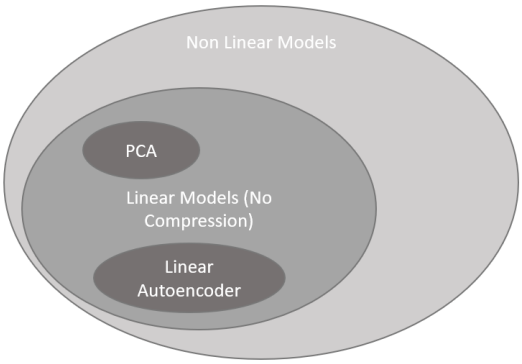


Figure 9 - Comparison Of Power Of Compression Techniques

Unfortunately, this model is relatively opaque, so the specific values of the participants that the model is using to understand predict the participants responses are not interpretable. Further models in this report perform worse on the test set, with the intention of genialising to the training set better.

7.2 PCA Compression

To alleviate overfitting concerns, an intermediate compression layer was placed between the raw embeddings and the value comparison. The PCA is the simplest method of compressing a latent feature space. The following results contain 2 dimensions of heterogeneity.

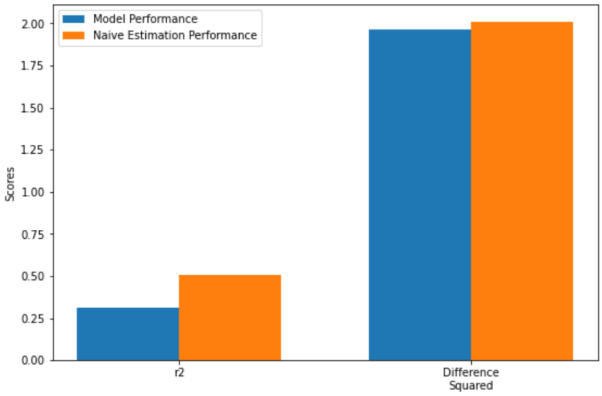


Figure 10 - Model Performance on Training Data

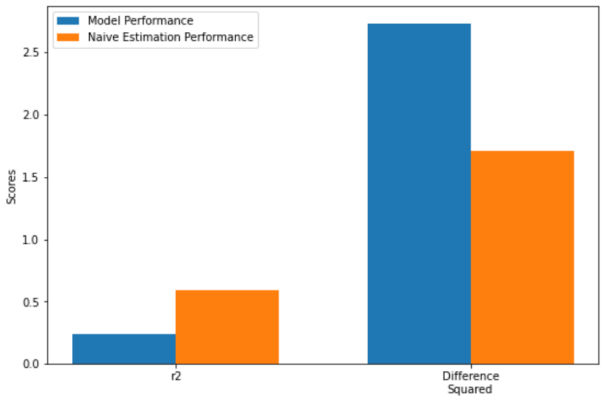


Figure 11 - Model Performance on Test Data

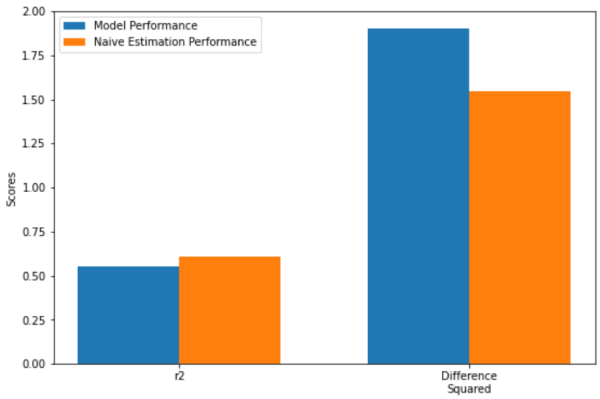


Figure 12 - Model Performance on Q22.2

Overall, the PCA model performed significantly worse than the model with no compression. This is likely because the strict reductions in the number of parameters in the model does not fully encode the full span of each sentence.

All question sets had a poor performance when compared to the base case. The training set, which ordinarily should perform much better than base case, performed worse than the base case. A lower r^2 of 0.25 and a comparable difference square lead to a poor model performance. The test set performed significantly worse than the training set implying very poor generalisability.

Since the above model maintained relatively low performance in both the test and training samples, the aggregate model must be underfitting. To allow the model more flexibility to fit the data, more dimensions of heterogeneity were used to give the aggregate model more flexibility.

7.3 PCA Compression 5 Dimension of heterogeneity

To improve upon the original PCA model, which performed poorly, more dimensions of heterogeneity were introduced. The following results used 5 dimensions of heterogeneity on the PCA compressed embeddings.

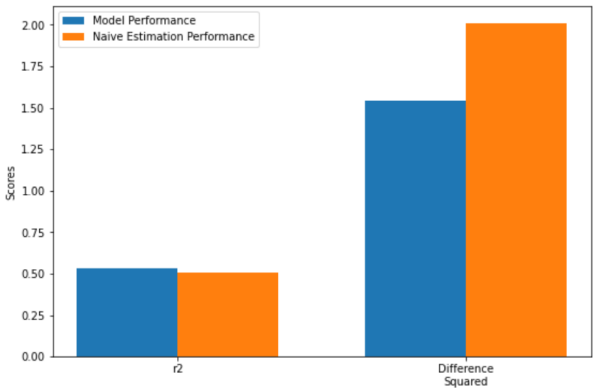


Figure 13 - Model Performance on Training Data

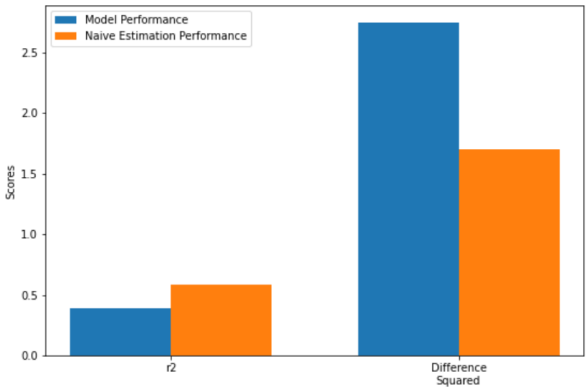


Figure 14 - Model Performance on Test Data

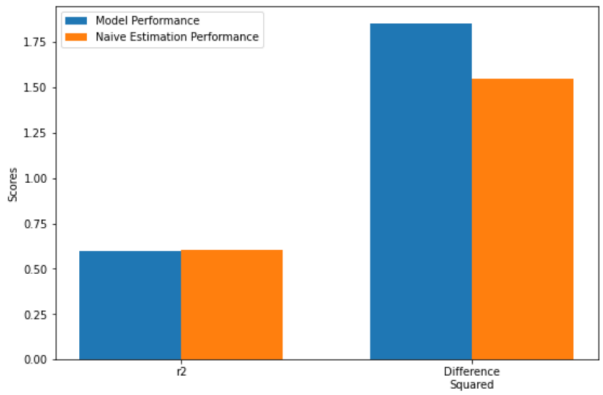


Figure 15 - Model Performance on Q22.2

Increasing the number of dimensions of heterogeneity to 5 dimensions allows for a more powerful model which performed much better than the standard PCA. By providing multiple dimensions of heterogeneity, multiple characteristics of the survey participants are calculated. The overall model's performance is improved by allowing more parameters of variability.

The extra parameters improve the model's performance to be better than the naive baseline in the training set. The baseline r^2 slightly increased by 0.02, and the difference squared decreased by 0.5. Improving the training performance also improved the test performance on Q22.2 to that comparable of the naive estimation. Unfortunately, this was not as effective as a no compression method.

Note, A grid search in terms of the optimal number of dimensions of heterogeneity was performed, and 5 dimensions performed the best for the PCA based models.

7.4 Premeasure

The premeasure technique revolved around comparing sentences to a large list of commonly used adjectives. This constrained the model to more conceptual information and resulted in the following performance. The following results were run using 2 dimensions of heterogeneity (for rational see section 8).

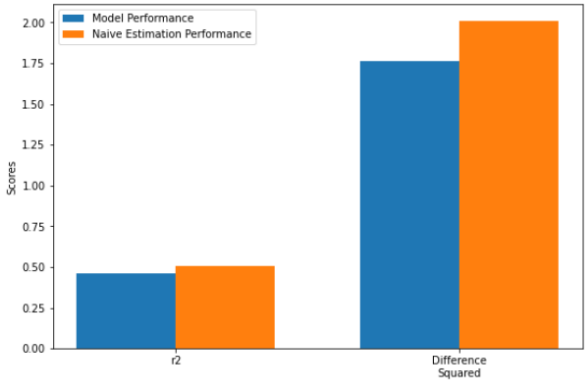


Figure 16 - Model Performance on Training Data

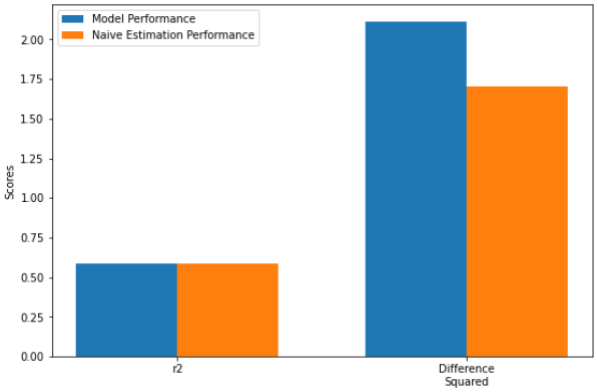


Figure 17 - Model Performance on Test Data

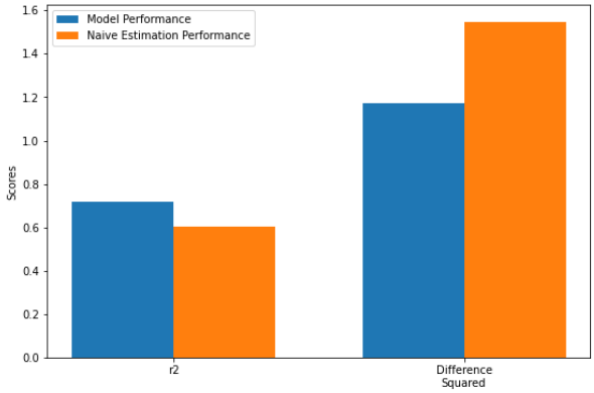


Figure 18 - Model Performance on Q22.2

Using a premeasured set of results constrained the model to more conceptual aspects. Measuring each sentence against a specific adjective reduces the dependency of the model on the structure of the sentences, and instead forces the model to focus on more conceptual aspects of each prompt. These conceptual constraints, allow for more meaningful models and better generalisation.

These constraints primarily improved upon the generalisability of the model. Compared to the no compression technique, the pre-measured results performed comparable with a much broader generalisability. Specifically, this model performs much better on a general training set than any of the PCA or no compression models. The test set r^2 was the same as the naive estimation, and the difference squared only increased by 0.27.

The limited information in this model reduced the precision in the training sets, but the comparatively high predictive power of this model indicates a strong trade off.

7.5 Auto encoder

In this approach a shallow auto encoder based neural network was implemented to reduce the dimensionality of the output. This is the highest parameter model and is the most likely to overfit. However, when viewing the output of the model the opposite appears to happen.

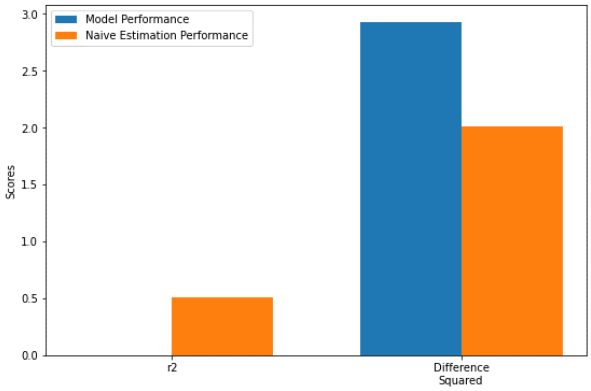


Figure 19 - Model Performance on Training Data

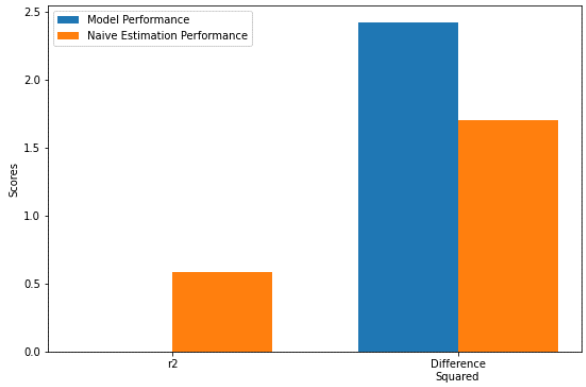


Figure 20- Model Performance on Test Data

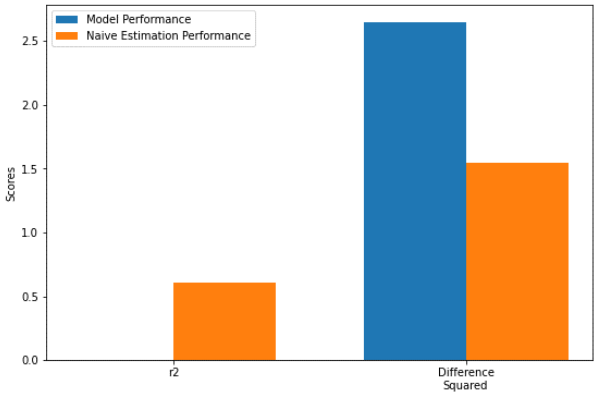
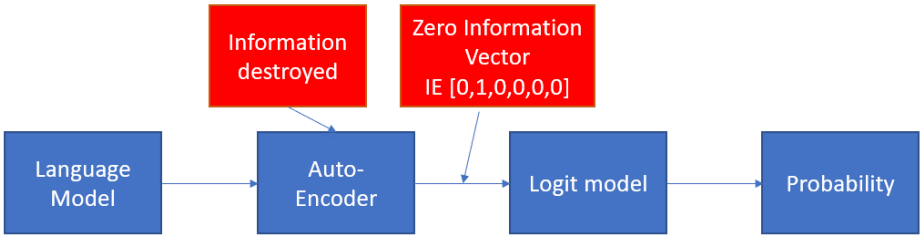


Figure 21 - Model Performance on Q22.2

Unfortunately, an unaugmented auto encoder did not reach the desired performance. The model immediately converged to an average response of somewhat agree. This performance was not exceeded despite the ample number of iterations for convergence.

The primary difficulty was that the auto encoder model “destroyed” the information contained in the sentences. This left no more information for “logit” proportion of the model to fit. This effect can be observed by viewing the outputs of the small neural network embedded in the aggregate model. The small neural network converges to a “zero information” vector which typically contains 1, (sometimes more) value that is fed into the remaining logit model. Since no more information is given, the remaining model can only approach an average of all participants. The entire model is seen below.



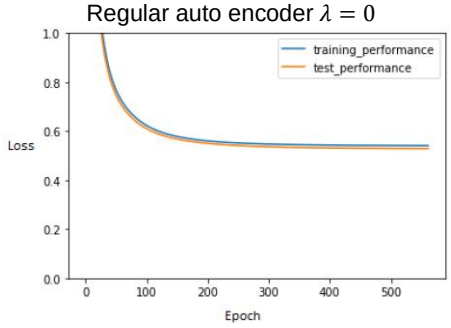
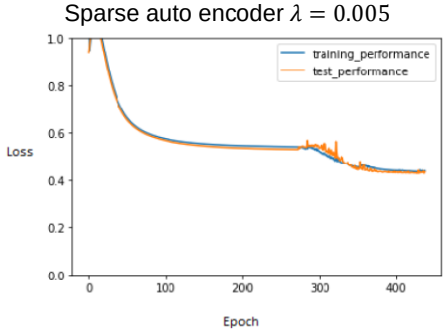
To improve upon these results a sparsity constraint is added as outlined in the.

7.6 Sparse Auto Encoder

The sparse auto encoder adds a penalty for large weights on a matrix. This penalty means that the model is discouraged from fitting parameters which destroy information unnecessarily and gives more opportunities for meaningful regression.

These improvements can be seen when viewing the training loss for a regular and sparse auto encoder shown in Table 9. The regular auto encoder immediately approaches the average and fails to improve. The sparse auto encoder eventually archives a higher performance.

Table 9 - Comparison of Loss for Auto Encoders with Various Sparsity Constraints

Regular auto encoder $\lambda = 0$	Sparse auto encoder $\lambda = 0.005$
	

The sparsity constraint also improved the problem of information being destroyed during the compression process. The intermediate outputs (below) clearly show the auto encoder extracted at least 2 dimensions of information (not 0 or 1):

Intermediate output: $[0 \quad 0 \quad 0 \quad 0 \quad 2.205 \quad 0 \quad 0 \quad 0 \quad 0.639 \quad 0]$

This gave the following results:

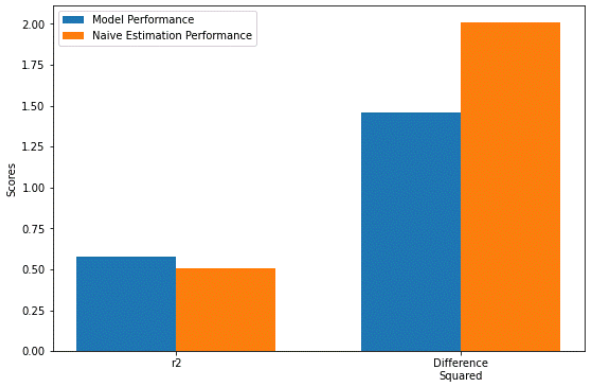


Figure 22 – Model Performance on Test Data

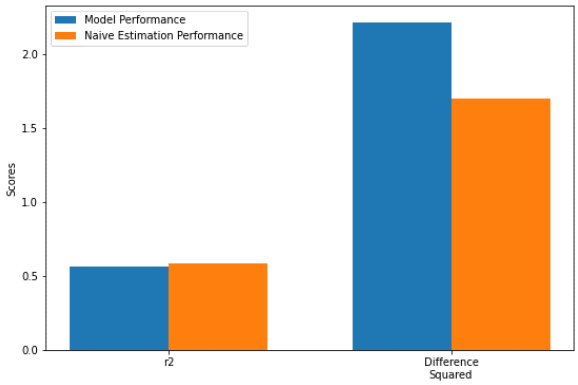


Figure 23 - Model Performance on Test Data

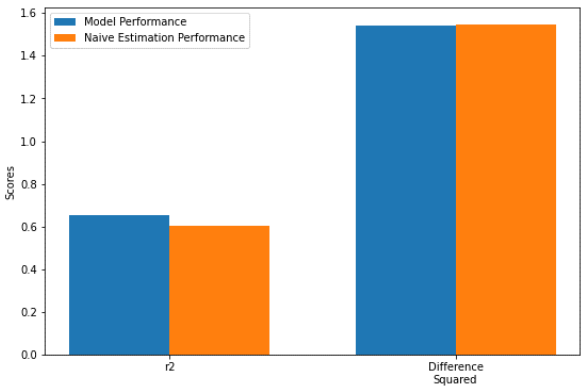


Figure 24 - Model Performance on Q22.2

The sparse auto encoder showed some of the most general results of the study. This is likely because the encoder portion reduced the sentence embeddings to 2 dimensions. These two dimensions were forced to be broad because

The sparsity constraints primarily improved upon the generalisability of the model. Compared to the no compression technique, the sparse results performed slightly worse with a much broader generalisability. This model performs much better on the training set than any of the PCA or no compression models. The test set r^2 was comparable to the naive estimation, and the difference squared increased by 0.4. Q22.2 maintained comparable to the naive model, although slightly improved on both difference squared and r squared.

8. Heterogeneity Results

To assess the models understanding, statistical analysis on the heterogeneity parameters was assessed. The no compression model was relatively opaque, especially with respect to the conceptual understanding. This indicates that these heterogeneity parameters may not be analysable to show specific concepts are preferred/disliked.

The Heterogeneity analysis was primarily performed on gender. This study primarily focused on gender, as it is a universal factor that was asked in the study. The greater male heterogeneity effect (Thöni & Volk, 2021) was also focused upon, as this can easily be analysed with an F tes. Given the specific heterogeneity implementation and the lack of interpretability for some models, the primary goal of this analysis is to observe higher heterogeneity of males then females.

To create the following graphs, the heterogeneity parameters that were calculated as outlined in 6.6 (taken as pA). Each survey respondents' parameters were extracted, and the distribution of these parameters are compared between males and females.

8.1 No Compression

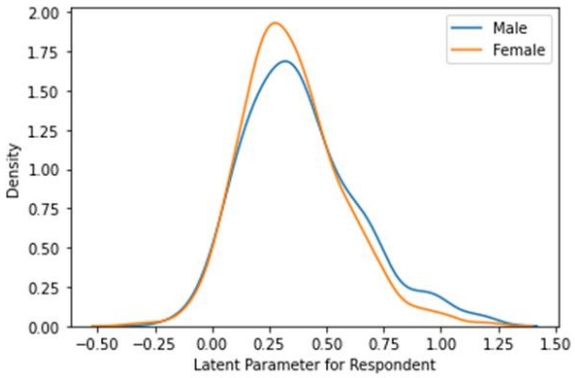


Figure 25 - Male vs Female First Latent Parameter

The T-test on these heterogeneity parameters was insignificant ($p = 0.2$). However, when comparing the variances of two reported genders, F statistic showed these two distributions showed a significant value of 10^{-9}, This is strong evidence that the model has discriminatory power and is interpreting some general conceptual patterns.

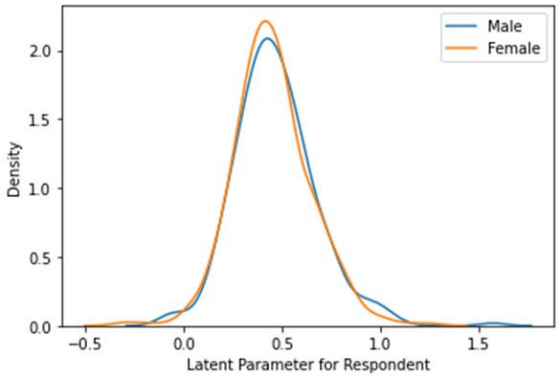


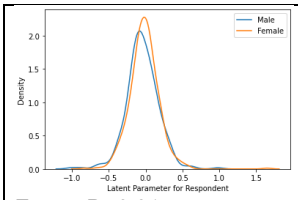
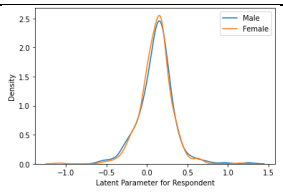
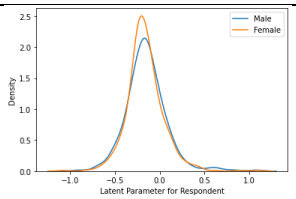
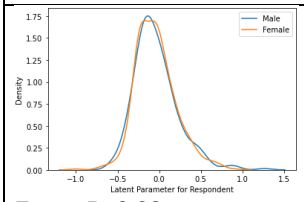
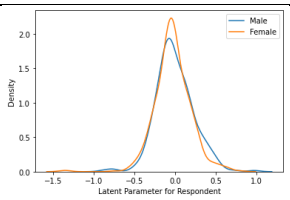
Figure 26 - Male vs Female Second Latent Parameter

The T-test for age a defining heterogeneity parameter for the showed that age was also a considerably power explanatory variable ($p< 10^{-11}$). Generally, even with uninterpretable results, heterogeneity is understood by the model.

8.2 PCA 5 dimensions

The heterogeneity analysis was applied to the latent parameters on the 5-dimensional PCA model. Since 5 separate distributions are being compared, a P-value less than 0.01 must be observed to be considered statistically significant (to alleviate P-hacking).

Table 10 - Collection of Heterogeneity Parameters For 5 Dimension PCA

 F test: P=0.01 T test: P=0.06	 F test: P=0.02 T test: P=0.61	 F test: P=10^{-4} T test: P=0.38
 F test: P=0.02 T test: P=0.78	 F test: P=0.14 T test: P=0.1	

Note that only 1 statistically significant heterogeneity parameters were only observed in 1 parameter in this subset of results. Generally, there was a slight trend of higher male heterogeneity, but this is not statistically significant such as was seen for the other compression techniques.

8.3 Premeasure

The heterogeneity parameters were compared for the premeasure parameters. These heterogeneity parameters were compared for male and female and presented below.

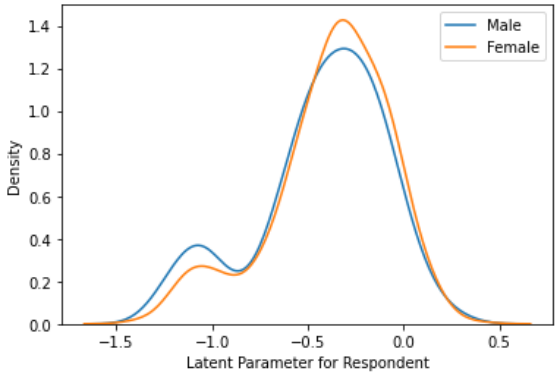


Figure 27 - Male vs Female First Latent Parameter

The T-test on these heterogeneity parameters was found to be significant ($p = 0.035$) although the base data is not normal which is an assumption of the T-test. When comparing the variances of two reported genders, F statistic showed these two distributions showed a significant value of 0.01, this is strong evidence that the model has discriminatory power between males and females.

The model also appeared to produce two groups of people, indicating two separate comparisons values by which the participants valued the set of questions. These two groups are encouraging as it indicates a deep understanding of the survey respondents.

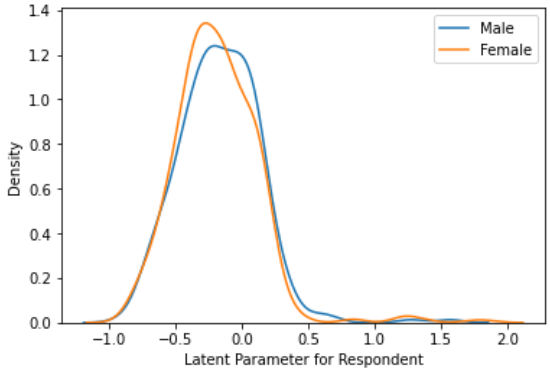


Figure 28 - Male vs Female Second Latent Parameter

The T-test on these heterogeneity parameters was insignificant ($p = 0.31$). When comparing the variances of two reported genders, F statistic showed these two distributions showed was also insignificant with a p value of 0.9.

8.4 Sparse Auto Encoder

The heterogeneity parameters for the sparse auto encoder were compered against each other for the male and female respondents.

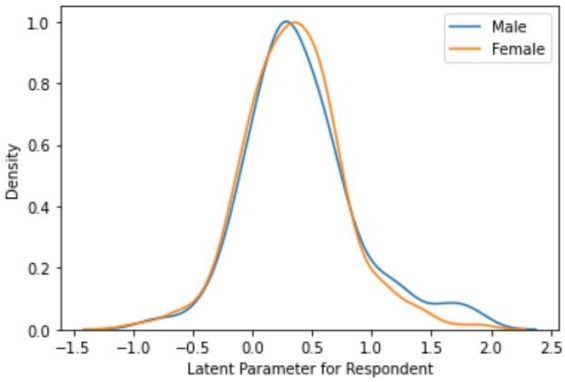


Figure 29 - Male vs Female First Latent Parameter

The T-test on these heterogeneity parameters was significant ($p = 0.043$). The F statistic for the above graph strong evidence of different variances ($p = 0.0041$). These statistics show significant evidence that the auto encoder discriminates between male and female on the first latent parameter.

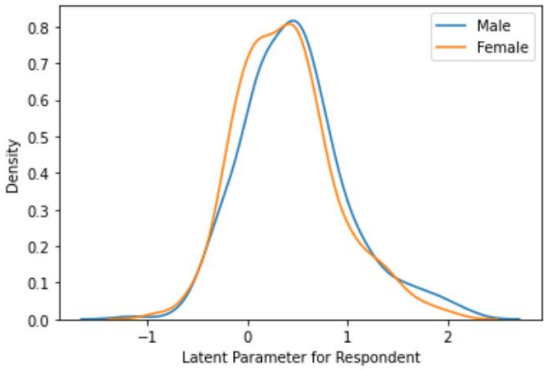


Figure 30 - Male vs Female Second Latent Parameter

Like the first latent parameter, there was significant evidence of different means ($p = 0.018$). There was little evidence of differences in heterogeneity ($p = 0.1$). The significant T test also implied that some of the differences between males and females are encoded in this latent parameter.

9. Interpretability

The technique outlined in 6.3 was applied to each relevant model. The resulting outputs of this analysis are shown below.

9.1 PCA

Table 11 - Adjectives with Highest Activations Of Compressed Layer

	Most significant	Second Most significant	third most significant
Positive	Sociable	Quarterly	Expensive
	Harmonious	Annually	Fat
	Pleasant	Monthly	Shoddy
	Happy	Yearly	Wobbly
	Jovial	Weekly	Lanky
Negative	Monthly	Disguised	Elementary
	Weekly	Illegal	Arctic
	Quarterly	shady	Revolving
	Annually	Unaware	Electric
	Dead	Questionable	General

The technique of principle component analysis provided some relatively interpretable values for each neuron activation. Although these measures did not appear to line up with values a respondent would care about.

The most significant neuron appeared to focus strongly on sentiment. With positive activations centring on classically positive words including sociable pleasant and happy. This is strong evidence that the language model interpreted the questions as spanning many different sentiments. The negation of this neuron (IE strong negative values) implied a focus on time.

The second most significant neuron appeared to focus primarily on time. The negation of this neuron appears to deal primarily with negative sentiment with words like “shady” and “unaware”. The model appeared to combine sentiment and “time relatedness” into the first neuron.

The third most significant activation is difficult to infer meaning from, but the presence of electric in the negative activation implies that this activation might indicate the questions relation electric vehicles.

9.2 Premeasure

Table 12 - Adjectives with Highest Activations Of Compressed Layer

	Most significant	Second Most significant	third most significant
Positive	Grounded	Accomplished	Definitive
	Unused	Wise	Whole
	Lost	Usable	Utter
	Neglected	Aware	Complete
	Abandoned	Able	Absolute
Negative	Overjoyed	Dead	Snoopy
	Big-hearted	Weekly	Kosher
	Tall	Sweltering	Sleepy
	Giddy	Silent	Criminal
	Jubilant	Snoopy	Sad

The pre-measured technique was the most interpretable result. There results appeared to line up with those that a respondent might care about/be influenced by.

The most significant neuron appeared to focus strongly on sentiment. With positive activations centring on classically positive words including s*ociable pleasant* and *happy*. This is strong evidence that the language model interpreted the questions as spanning many different sentiments. The negation of this neuron appeared to indicate negative sentiment, leading to this value being exclusively focused sentiment of the sentence.

The second most significant neuron is slightly less conclusive. There is considerable evidence that this measure indicates status. The words that peak this value include *accomplished*, *Wise,* and *Able.* The negation of this does not appear to be oriented around status. The presence of 3 words relating to the status in this indicates some focus on status.

The third most significant activation could possibly be interpreted as definitiveness of the neuron. High ranking words include *Definitive, Whole,* and *Absolute.* Although slightly less conclusive than the other activations, there is still some evidence that this is an interpretable value.

9.3 Sparse Auto Encoder

Table 13 - Adjectives with Highest Activations Of Compressed Layer

	Most significant	Second Most significant
Positive	Kosher	Recent
	Antique	Capital
	Zigzag	Early
	Motherly	Front
	Medical	Annual
Negative	Gregarious	Zigzag
	Strident	Limping
	Dense	Loathsome
	Bulky	Lonely
	Medium	Lost

Unfortunately, very little interpretability was seen in this model. Even though only 2 values for beta were fitted. The most significant activation had *Kosher, Zigzag,* and *Motherly* as the most significant positive activation. The negative activations did not have any meaningful patterns either. This indicates that the techniques of sentiment analysis applied to this network did not afford much interpretability to the neural network-based model.

10. General Evaluation and Discussion

All results had some level of generalisability to out of domain training error. The generalisability indicated that all models – except for the auto encoder – used the language model embeddings as significant information to improve the predictions beyond the naive model in some capacity. Except for the auto-encoder approaches, all compressed models had some degree of interoperability. The most significant factor the interpretable models observed was overall sentiment. Patterns in heterogeneity were also found by performing analysis on intermediate feature spaces of the model, these patterns reaffirmed the well-known phycological result of males having a higher heterogeneity in almost all fields (Thöni & Volk, 2021).

An effect that was observed amongst all the results in this report was the trade-off between having a generalisability and precision predictions. The genal trend amongst all interventions was the higher the performance in the training set, the lower the test performance of the test set. This follows from overfitting data where the model fits spurious relationships amongst the data that do not indicate any significant trends. As various constraints and compression methodologies were introduced, the model was restricted from fitting these relationships.

10.1 No compression Predictive Power

The model with the most precise predictions, specifically when ignoring overfitting concerns, was the no compression model. The no compression had the most explanatory power over the training data and some out of domain training questions. In the training data, the r squared increased to 0.56 and the difference squared metric was reduced to 1.32, indicating that the model maintained a significant amount of explanatory power. The model also maintained limited explanatory power of novel questions, specifically Q22.2 maintained an r squared of approaching 0.8 and a difference squared close to 1.01. This model struggled when compared against the larger random set of removed questions. The random removed questions had the worst performance of all models explored in this report. This indicates that the model has a high sensitivity to specific nature of the questions it generalises too.

The no compression model had a relatively large number of parameters (4164 parameters). The parameters mostly used fitting heterogeneity since the input dimensionality are relatively high. The large number of parameters is likely the culprit for the overfitting behaviour of the model. Conversely, the large number of parameters gave the model a lot of explanatory power allowing it to make precise predictions of the training data set.

10.2 Auto encoder Predictive Power

The auto encoder had little predictive power as the model predicted the average response of all respondents. The sparsity constraint was added to attain better modelling performance.

10.3 Sparse Auto encoder Predictive Power

The second most performant model was the sparse auto-encoder model. The r squared attained 0.55 and the difference squared of 1.51. This model had comparable performance to the no-compression model and generalised to the test data set much better.

The simple 2-layer neural network had the ability to interpret some patterns and translate nonlinear trends into the logit model. This did not correspond to higher overall performance and more overfitting, which would be expected of the less constrained model of a neural network. In section 0, the outputs of the neural network were investigated to understand why this comparatively weak explanatory power was observed in the model's performance.

The outputs of the sparse auto-encoder were observed as shown in Figure 31.

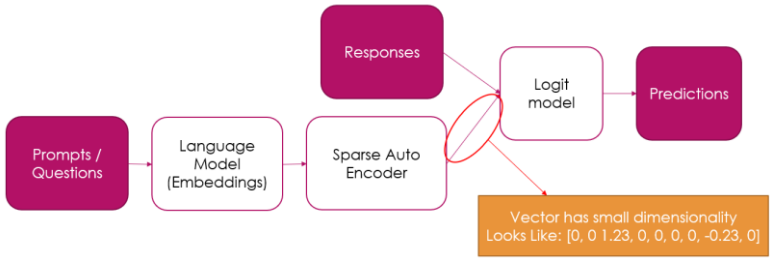


Figure 31 - Sparse Auto Encoder Reduction In Dimensionality

The auto encoder reduced the information found in the intermediate layers down to 2 dimensions. This reduced the explanatory power of the model to 2 dimensions, IE only 2 betas in the logit model. In this model the final logistic equation would be:

$$U = \beta_1 M_1 + \beta_2 M_2$$

This occurs because auto-encoder based models will only fit trends and patterns that it has the capacity to predict. In this implementation the model did not find any trends or patterns that could be explained by more than 2 variables. This is encouraging as it shows why the model refused to fit spurious trends and subsequently overfit less.

This result was also extremely sensitive to the ridge regression coefficient lambda. If the coefficient was too small/large the intermediate layer would degenerate into 1 or even 0 significant coefficients. A lambda of 0.08 found to create autoencoders which had at least 2 intermediate values which were active feeding into the logit model.

The degenerate output of the sparse auto encoder to 2 dimensions does not necessarily imply that the only trends in this data that can be found only span 2 dimensions. The high training performance of the no compression model indicate that many more correlations exist in the training. The degeneracy of the auto encoder is the balance of the sparsity penalty and the data performance.

The degeneration was found to be overall useful as it restricted results and reduced the number of solutions to broad patterns.

10.4 Predictive Power of PCA Models

The PCA based models did not maintain a high degree of predictability. This is likely because the PCA compression is “blind” to the responses. The values that the PCA model was extracting from language model had no statistical reason to heavily correlate with the responses. The PCA compression would only work if This feature was also present in the premeasured model, but that model had maintained relatively high performance as adjectives tended to line up with human decisions.

10.5 Predictive Power of Premeasure Model

The overall performance of the pre-measured results was quite high, especially considering it relatively small propensity towards overfitting the results. With the training r square of 0.48 and a difference squared of 1.75. The Training data did not reach an r squared larger than

the naive estimation, this is likely due to an in-optimal stopping time, as early stop technique is used to avoid overfitting. The generalisability of this model implies that restricting the measurements to conceptual information, rather than all possible information, found more significant trends and correlations in the data. This is further supported by the human interpretability by the neurons.

10.6 Interpretability

The detailed analysis of the results outlined in section 9 is shown below. The no compression model was not considered to be an interpretable model and so detailed analysis was not performed.

Sparse Auto encoder

For the Sparse Auto Encoder, when observing the neuron activations on a set of adjectives, the overall outputs (section 9) were not very interpretable. The auto-encoder has an ability to make complex, non-linear correlations in the data to produce a set of measurements. It is possible that these nonlinear correlations cause the sentiment analysis techniques model to perform poorly. This is because the single word "sentences" are very different from the multi word questions created in the model. Non-linear trends tend to be particularly poor at extrapolating data. An abstract graphical representation of this is found in Figure 32.

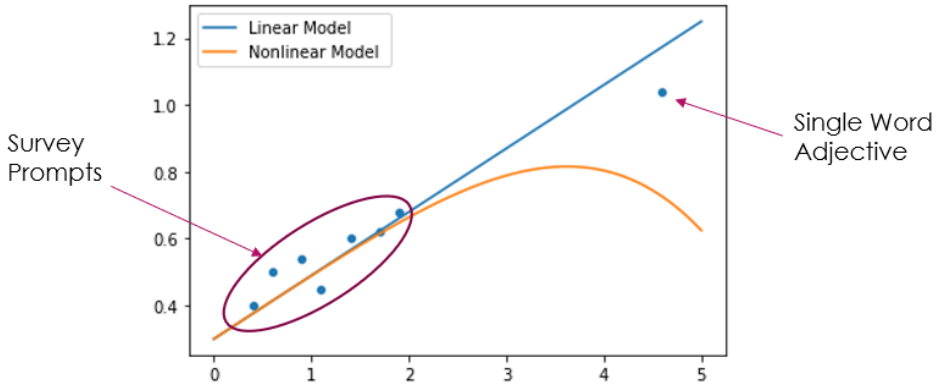


Figure 32 – Non-Linearity Effect on Extrapolation

This problem is generally exasperated by fitting non-linear models in higher dimensions. This means that when the model attempts to summarise a model with a single word, it is dominated by spurious correlations that fit nonlinear terms.

Pre-Measure

The most interpretable model was the pre-measured model. This is because this is one of the more restrictive compression techniques where it forces each of the beta coefficient to correspond to a sample adjective. This forces the model to focus primarily on the conceptual information as opposed to other aspects that might be encoded in the word model. This led to beta coefficients that strongly correlate to the representative adjective of the sentence.

This model also led to two groups of respondents. This is a very encouraging result as it indicates that the model understands and separates two groups of people. Some respondents in the survey cared about different aspects of the questionnaire then others.

The preference for which group the participants fell in was found to be partially explainable by socioeconomic factors such as male/female. Unfortunately, these groups are correlated with an “abstract” parameter, so interpretation of the meaning behind these groups would require more detailed analysis.

PCA

As outlined in 9.2, the PCA did not provide excessively interpretable results. This is due to the simple compression methodology not having information of the survey respondents. The only patterns the PCA method found in the text itself and were less relevant to the responses.

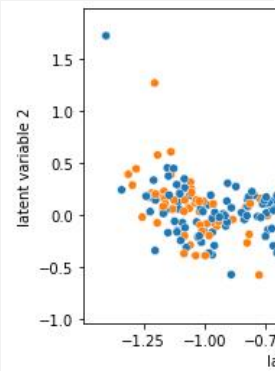
10.1 Heterogeneity

All models with 2 heterogeneity parameters attained a statistically significant discriminatory power to differentiate male and female behaviours. All models found that males had higher heterogeneity of latent parameters then females. The higher heterogeneity of males is confirmed by male variability hypothesis as stated in the literature.

The only model that did not find statistically significant differences was when 5 heterogeneity parameters were used. The lack of statistical significance appears to be because any effects are spread out over the 5 fitted parameters. Each individual parameter did observe statistically insignificant effect of greater male heterogeneity.

The most interesting result was found in the Premeasure (section 8.3) where one of the latent variables grouped the behaviour of the participants into 2 modes of behaviour. This is an ideal result in an unsupervised learning task, as it implies that the model has a strong enough understanding of the data set to divide them into 2 modes of behaviour. Possible further work could be to investigate these two modes and interpret what the primary values of these groups.

Overall, the model quite successfully modelled heterogeneity within the survey respondents.



10.2 Sensitivity of Analysis

The results attained from all methodologies were relatively sensitive to many variations. These factors include:

- Compression technique
- Specific out of domain questions
- Random initialisation
- Heterogeneity Parameters

And are explored below.

Compression technique

The compression techniques used in this report were the primary intervention to improve performance and generalisability. Depending on the method of compression used, the model either servilely overfit (such as the results seen in no compression) or drastically reduced the explanatory power of the model (such as the results seen in PCA).

As stated above, the general trend was observed that peak performance was traded off with generalisability. This indicates that, with further engineering and investigation, the model which performs in the most generalisable way could be found to predict an unseen question. Ideally a hypothetical final model would actively change its parameters given an unseen question. An adaptive compression technique is out of scope of this investigation; however, they may be effective at mitigating some of this high sensitivity.

Test/unseen questions

The model was found to be very sensitive to the out of domain questions it was attempting to predict. The high sensitivity found in the test set of questions was due to them not being designed for analysis using logistic regression. The questions were deigned to have little overlap, despite this the model still managed to generalise some broad patterns in the questions including sentiment and possibly status as outlined in section 8. Although a high sensitivity was found observed in the unseen questions, the models still were able to generalise.

Random initialisation

As is standard in machine learning, the initial weights of the in the model were randomly initialised. This led to the overall model performance not being identical when the analysis is rerun.

The model with the highest sensitivity to initialisation was the sparse auto encoder model. This model would sometimes degenerate to a single dimension when attempting to fit the data. The initialisation of the model parameters influenced if 2 dimensions would be depending on the initialisation. Each model was run multiple times to ensure that the best performance was attained. Apart from the sparse auto encoder, all other models were relatively robust to specific initialisation, with final performances changing by less than a percent.

Heterogeneity parameters

As can be seen from observing the PCA case, the number of heterogeneity parameters generally improved performance on the test set. More heterogeneity parameters improved performance because it gave more explanatory power to the model. However, this also raised the overfitting concerns of the model. Due to the results of the 5-dimensional PCA (section 0) more than 2 heterogeneity parameters generally were not found to be significant enough

to use in most models. The auto-encoder models would not provide more significant results with more heterogeneity parameters since the compressed space was as low as 2 parameters.

10.3 Observations During Model Development

A difficulty that was found during the implantation and analysis of the model was concrete definition of a predictable unseen question. An argument could be made that the first model, with no compression, achieving the best results as it fit the training data the best, and fit to the special case (Q22.2). The key issue involves weather the questions removed from the data set are “modellable” with the given questions. A hard limit in the effectiveness of modelling methodology exist in terms of how much the models overlap. For example, the responses for “people are generally trustworthy” and “I am familiar with the work of smash mouth” would have very little correlation. The overall methodology of the model requires conceptual correlation between the questions. Because of this undefined conceptual overlap, the ability of the model to fit unseen questions varies depending on the specific removed subset.

Most the models applied in this report were constrained to the linear transformations of the input parameters. This effect was mostly found because language model embeddings are designed to translate human concepts into a representation of vectors embedded in an abstract feature space. The embeddings optimised by a “desired norm”, which is focused on conceptual distance between two sentences or phrases.

The results found in this report indicate that the relationship between embeddings and decisions are mostly linear. This can be seen with the high performances of the linear models. For the questions surveyed in this report, the value/utility could be represented as some weighted model of the conceptual understanding represented by the questions. Non-linear models also found some correlations but did not significantly outperform the linear.

Model power compared to number of parameters

The naive results highlighted in this report simplify all responses made by a respondent into 1 single parameter. This single naive parameter effectively encodes how agreeable a participant is when given a prompt.

Most models in this report had some base number of parameters and an additional number of parameters per person. This relationship is expressed in Table 14:

Table 14 - Comparison of Number Of Parameters

Naive Estimation	Sentence Embedding Model
$NP = R$	$NP = H_d \times R + C$

Where:

NP: Number of parameters for a given model

R: Number of unique respondents in survey

H_d: Number of Heterogeneity parameters calculated per unique respondent

C: The number of parameters included in the model independent of the number of survey participants.

As can be seen, the models proposed in this study always maintain more parameters, and thus have more explanatory power over the possible participants. So long as overfitting concerns are not exceeded, the more parameters do appear to add to the predictive capability of the model.

High Predictability of Question 22.2

Although an exhaustive search was not performed amongst all possible test/training partitions of the data. In the test set, the best performing question was Q22.2. For all above results Q22.2 was removed from the training data, it still performed significantly better than the training data for many questions. Typically, data in the test set is expected to perform worse than the training set. There are multiple possible explanations for the unexpectedly high performance of this question including similar encoding to another question, cumulative generalisation from the entire model and the question varying along sentiment lines. It is likely that the overall performance of this question can be explained by a combination of these factors.

Question 22.2 potentially had better explanatory power because it had a high heterogeneity in of its responses. The statistical measures decided upon in this experiment are primarily focused on predicting differences in how a respondent answers a question. In the extreme case If the true response on a question is the same amongst all respondents, the model will over-discriminate between the survey respondents.

Commented [LP17]: Can we get evidnce?

10.4 Further work

To keep the body of work within a reasonable scope for delivery, multiple concessions were made to maintain a deliverable project in budget. Potential further work and investigations are outlined below.

Larger/different language models

The language model adapted for this report is derived from 2018 BERT by Facebook (Vaswani, 2017). Although this model was state of the art when it was released, further work has been explored by GOOGLE, FACEBOOK, OPENAI. Typically, the dimensionality of the embeddings of these models is larger than the approximate 700 of the BERT models. More dimensions would further exacerbate the difficulties explored in this report of developing compression techniques.

More recent language models also tend to be more costly to run. The BERT model used in this report small enough to exist entirely on local ram. More modern, larger language models require large cloud-based software which typically costs a fee per query. To save on cost and reduce the scope to a more focused set of techniques, a single language model was used for all analysis on this paper.

Further work could compare more refined language models with this technique and investigate the improvements to accuracy and generalisability.

Custom survey data

An important discovery that was observed through the analysis performed in this report was the relative sensitivity of the model to different survey questions. The training set contained a wide variety of performances ranging from no predictive ability to an ability much better than the comparison methodology.

During this study, various hypothesised methods were proposed to predict and/or construct a novel questions predictability. These methods were not fully investigated but are explored below.

The research from in generalisability of deep neural networks is applicable to this problem. Various methodology has been proposed in recent years to assess a machine learning model. The models presented in this report are much shallower but the techniques for deep neural networks exist (Guan, Analysis of Generalizability of Deep Neural Networks Based on the Complexity of Decision Boundary, 2020) are expected to generalise to them. These methodologies are primarily concerned with the shape and roughness of the decision boundary of the model. Unfortunately, with the current implementation, it is difficult to define a representation of the input space in which to draw the decision boundary. Text classification is a high dimensional problem where highly different inputs in feature space may be encoded similarly in an intermediate layer. An example in a language model is the sentences “A fast car can help you” is more like “you need a fast car for help” than “a fast workout can help you”, even though the latter has a much more similar characters to the original than the former.

Presumably at some intermediate layer of the entire model there exists some abstract feature space which contains an interpretable and useful decision boundary. Finding a decomposable language model and investigating multiple representations of abstract features was considered beyond the scope of this investigation.

Another potential method of screening question sets which are susceptible to this kind of analysis is to take the hessian of the output from an intermediate feature space. This may be especially effective when the remaining model consist entirely of linear transforms, such as the *PCA* and pre-measure. The hessian indicates the rate at which the model diverges from a sample. This can be expressed below:

Given a sentence S and a model decomposed into the composite of two smaller models M_1 and M_2, (IE the cumulative model for $M_1 \circ M_2(S) = U$):

$$I = M_1(S)$$

$$U = M_2(I)$$

Then, using a multi-dimensional Taylor expansion, the effect of error on the divergence of the solution from a linear space, can be calculated:

$$U = M_2(I) + \epsilon \times \nabla M_2(I) + \frac{1}{2}(H(I + \epsilon\theta)\epsilon) \cdot \epsilon + higher\ order\ terms$$

Where $\|\theta\| = 1$

Assuming the solution is represented as a linear function of M_2 then the overall error must be proportional to the hessian (this effect will likely hold without this assumption). Then the model’s ability to generalise the training set to an unseen question can be modelled if there exists a training question (T) where an unseen question (U).

$$|M_1(T) - M_1(U)| \leq desired\ error\ derived\ from\ hessian$$

Unfortunately, the language models used in this report were “black-box”, where access to the intermediate layers were not accessible. This methodology could potentially be pursued in further work.

A simple methodology was briefly explored involving comparing the cosine distance of the embeddings for the test and the training set. This type of analysis did not provide any significant results.

Composition of compression techniques

Another potential methodology improvement that could be explored in further work involves the composition of many of the compression techniques proposed in this report. Different modelling methodologies maintain different advantages/disadvantages. For example, the pre-measure technique is specifically effective at ignoring the syntactic structure of a text excerpt, and the PCA methodology reduces the probability of overfitting. Combining them together, could result in a preferable result compared either technique on its own.

This methodology was not deeply explored in this report, primarily because of the excessive search space it would open the report scope too. A future report could explore the possible improvements to be had by combining more compression techniques.

The effectiveness of the compression techniques may also be used to evaluate the efficacy of the model. A high compressibility indicates that the model decomposes the question set into a distilled subset. The effectiveness of the models may be able to be assessed by the robustness and universality of the compression techniques.

Additional regression statistics

Due to the methodology, there does not exist any additional statistics around a best fit in a model (IE confidence intervals, standard deviation of each variable). A pytorch implementation required because backwards propagation of gradients to fit the best compression methods. In future work, additional analysis could be applied to the betas within the model to derive the statistical significance according to Xu's paper (Xu, 2005).

The models presented in this report can be classified as using both a machine learning and a logit methodology. In machine learning, it not recommended to remove intermediate layers based on statistical significance. Further research is needed to assess the validity of these additional regression statistics.

Hybrid modelling

Realistically this modelling methodology does not maintain a complete superset of an ordinary logit model. For example, the sentence "I would pay $1500 a year for a parking spot near work", would not be efficiently discriminated against the question "I would pay $50 a year for a parking spot near work" as language models typically struggle to assess numerical meaning.

A more useful implementation of this model involves a hybrid approach with a more traditional logit model. Hybridising the model by including traditional logit values could be achieved in the following ways:

1) The specific measures such cost or time could be concatenated to the vectors after compression.
2) A traditional logit model can be constructed, using the utility output from the language-based models. This has the advantage returning statistical significance for various betas.
3) The two models can be encoded separately, and any intersecting utility model can be removed using statistical analysis on any correlating betas between the two models.

Each of the methodologies have specific advantages/disadvantages that may be preferable depending on the specific context when hybridising the two models. Unfortunately, this work would require a new, larger set of questions/prompts to fit a hybridized model.

Normalizing data

Another potential improvement to the model is to pre-process the data to account for general enthusiasm of each respondent. As is shown by performance of the "naive" model, the proportion of the previous survey responses that a particular respondent has agreed with is a strong predictor weather the respondent will agree to a new question. Since this bias is easy to model for potential improvements could be made by normalising the data in a pre-processing step to allow the models to focus on the semantic information embedded in the text.

Other sparsity constraints

It is possible that other sparsity constraints could be included/used for the sparse auto encoder model. Sparsity constraints increase the performance of the model by placing a penalty on weights with the square of each weight's magnitude. This penalises the model for "destroying" information by penalising the model of having exceedingly large weights. For example, taking the sparsity constraint.

$$total\ loss = fitting\ loss + \sum_{i}\sum_{j} w_{ij}^2$$

Then the weights matrix could not degenerate into high values and low values:

$$W = \begin{bmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{m1} & \cdots & w_{mn} \end{bmatrix} = \begin{bmatrix} 10^3 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 10^3 & \cdots & 0 \end{bmatrix}$$

Since this constraint would punish the matrix for the large values. Other sparsity constraints exist such as the $L1$ norms, infinity norms, and various decomposition. These sparsity constraints can also be tried to achieve better compression of the question set.

10.5 Future uses

With further research the methodology proposed in this study could potentially be used for many future uses. A brief list of potential uses is listed below.

The predictive capabilities of the model could be used to generate fully synthetic survey questions responses. Training the model on significantly larger data sources with a variety of questions in a specific domain may produce a robust enough model to generally use the predictions without the need to conduct a survey. This potential future use could have vast economic benefits in many different fields where public opinion is important, and its collection is costly.

More development could also allow for detailed analysis to the reasoning behind the respondent choices. The ability of these techniques to find broad patterns in a questionnaire are powerful and could be used to further analyse any existing data set.

The results for heterogeneity from the premeasure approach (section 8.3) also indicated that the techniques proposed in this report can group respondents into unsupervised groups. With more refinement of the methodology, 3 4 or 5 classes of people with different fundamental reasonings behind there behaviour could be discovered, allowing for more detailed analysis of stated preference data.

Conclusion

The modelling methodology of using language models for generating various betas for logistic regression gave strong results. This methodology enabled interpretable models with some domain of explanatory power over unseen questions.

The language model-based approach appeared to understand and extract meaningful patterns in the question set for modelling participant responses. In most cases these language-based model maintained additional performance over a base model which failed to encode any data about the sentences.

Language based techniques were very sensitive in terms of their generalisability to unseen questions. In all iterations of the language-based logit model, the results did not generalise to all unseen questions. However, the model did show a strong ability to model a subset of unseen questions. This indicates that the language-based model maintains explanatory power that is somewhat general and robust.

The language-based models had some interpretability. By observing the activations of various adjectives that are fed into the model, strong evidence was given that multiple human interpretable trends were observed in the data. These results implied that the overall sentiment of a prompt has an impact on likelihood of the respondent agreeing. Possible further correlations were found in as “implied status” and “definitive statements” in a questionnaire affected how people responded to the prompt. These insights were observed without giving the model a pre-defined set of models to compare against.

The outer product decomposition methodology that was used for modelling heterogeneity on the outputs from the language model showed strong, statistically significant differences between male and female respondents. This gave strong evidence that the conceptual information expressed using this methodology differed between socio-economic parameters. The unsupervised grouping behaviour of the model further supported the model’s ability to understand trends in heterogeneity.

Overall, the combination of language models with typically un-modellable set of questions provided significant insights and more predictability in terms of survey responses. These models provided new insights and allowed for interpretability. The study successfully demonstrated language models’ ability to predict stated preference data.

11. Bibliography

Ash, E. (2021, December 17). *Hoover Institution Workshop On Using Text As Data In Policy Analysis*. Retrieved from Youtube: https://www.youtube.com/watch?v=oJa1wco2FxQ

Cristina, S. (2021, October 30). *The Transformer Attention Mechanism*. Retrieved from machinelearningmastery: https://machinelearningmastery.com/the-transformer-attention-mechanism/

Gattas, J. (2016). An Example Citation. *Journal of Fake Sources*, 12-25.

Godoy, D. (2018, November 22). *Towards Data Science*. Retrieved from Understanding binary cross-entropy / log loss: a visual explanation: https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a

Gruber, M. (2021, June 10). *How to Perform Ordinal Regression / Classification in PyTorch.* Retrieved from Towards Data Science: https://towardsdatascience.com/how-to-perform-ordinal-regression-classification-in-pytorch-361a2a095a99

Guan, S. (2019). Analysis of Generalizability of Deep Neural Networks Based on the Complexity of Decision Boundary. *Machine Learning*, 142-156.

Guan, S. (2020). Analysis of Generalizability of Deep Neural Networks Based on the Complexity of Decision Boundary. *19th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 14-17.

Hugsy. (2022, June 16). *english-adjectives.txt.* Retrieved from Github: https://gist.github.com/hugsy/8910dc78d208e40de42deb29e62df913

Jan Ketil Arnulf, K. R. (2014). Behaviour, Predicting Survey Responses: How and Why Semantics Shape Survey Statistics on Organizational. *PlOS ONE*.

JERRY HAUSMAN, D. M. (Econometrica). SPECIFICATION TESTS FOR THE MULTINOMIAL. *Econometrica*, 1219-1239.

JORDAN, J. (2018, March 19). *Autoencoders.* Retrieved from Jeremy Jordan: https://www.jeremyjordan.me/autoencoders/

Kazemnejad, A. (2022, 4 13). *Transformer Architecture: The Positional Encoding.* Retrieved from kazemnejad.com: https://kazemnejad.com/blog/transformer_architecture_positional_encoding/

Oxford Languages. (2022, April 30). *Break*. Retrieved from Oxford Languages: https://www.google.com/search?q=break&rlz=1C1RXQR_en-GBAU986AU986&sxsrf=ALiCzsZOCC1Qr5Je4RyZuUPgaQooRak-qw%3A1651935322429&ei=Woh2YuPdGYzCz7sPxO-PuAo&ved=0ahUKEwjj8tbQ0s33AhUM4XMBHcT3A6cQ4dUDCA4&uact=5&oq=break&gs_lcp=Cgdnd3Mtd2l6EAMyBAgjECcyBAgjECcyBAgjE

Pal, S. (2018). OuterSPACE: An Outer Product Based Sparse Matrix Multiplication Accelerator. *2018 IEEE International Symposium on High Performance Computer Architecture* . Vienna: HPCA.

Pan, C. (2020). Towards zero-shot learning generalization via a cosine distance loss. *Neurocomputing*, 167-176.

Rajamohan, S. (2018, September 9). *Word2Vec in Pytorch - Continuous Bag of Words and Skipgrams*. Retrieved from Gitlab: https://srijithr.gitlab.io/post/word2vec/

Riva, M. (2021, April 24). *Word Embeddings: CBOW vs Skip-Gram*. Retrieved from Baeldung: https://www.baeldung.com/cs/word-embeddings-cbow-vs-skip-gram

S, B. (2018, 12 06). *Language Models and Contextualised Word Embeddings.* Retrieved from davidsbatista.net: https://www.davidsbatista.net/blog/2018/12/06/Word_Embeddings/#:~:text=Word%20embeddings%20can%20capture%20many,into%20consideration%20even%20language%20models

Savage, J. (2019, March 18). *The logit choice model*. Retrieved from khakieconomics: https://khakieconomics.github.io/2019/03/17/The-logit-choice-model.html#:~:text=The%20logit%20model%20of%20choice%20models%20the%20latent%20fixed%20utility,Gumbel%20distribution%20with%20fixed%20variance.

Soma, j. (2022, April 25). *Intro to Word Embeddings*. Retrieved from investigate.ai: https://investigate.ai/text-analysis/word-embeddings/

Steffen Eger, A. R. (2019, June 4). *Pitfalls in the Evaluation of Sentence Embeddings.* Retrieved from Cornell University: https://arxiv.org/abs/1906.01575

Stewart, M. (2019, April 15). *Comprehensive Introduction to Autoencoders.* Retrieved from towardsdatascience: https://towardsdatascience.com/generating-images-with-autoencoders-77fd3a8dd368

Themantic. (2022, 4 25). *Sentiment Analysis: Comprehensive Beginners Guide*. Retrieved from Themantic: https://getthematic.com/sentiment-analysis/#:~:text=Sentiment%20analysis%20looks%20at%20the,can%20benefit%20from%20sentiment%20analysis

Thöni, C., & Volk, S. (2021). Converging evidence for greater male variability in time, risk, and social preferences. *PNAS*, 118-141.

Tomas Mikolov, K. C. (2013, January 16). *Efficient Estimation of Word Representations in.* Retrieved from Cornell university: https://arxiv.org/abs/1301.3781

Vaswani, A. (2017). Attention Is All You Need. *Computation and Language*, 55-70.

Xu, J. (2005). Confidence Intervals for Predicted Outcomes in Regression. *The State Journal*, 537-559.

Zhang, Y. (2022, May 3). *An Unsupervised Sentence Embedding Method by.* Retrieved from aclanthology: https://aclanthology.org/2020.emnlp-main.124.pdf

Appendix A – RACQ MAAS Survey

Question	Sentence
Q16.2	It is easy for me to travel to and from public transport (e.g. stops/stations are close by or have parking or good access via footpaths/bike paths).
Q16.3	I would cycle more if I had an e-bike (electric bike) which reduces the physical effort required.
Q16.4	I would cycle more if I could hire a normal or e-bike near my key origin and destinations.
Q16.5	I am happy to change modes a few times during my journey – e.g. from a bus to a train, or car to bus.
Q16.6	I am happy to walk a short distance (up to 10 minutes) to reach my transport.
Q16.7	I would use public transport more if it picked me up and dropped me off right near the start and end of my trip (less than 2 minute walk).
Q17.2	I would use a shared taxi or shuttle bus if it was cheaper than taking a private taxi.
Q17.3	I would consider not owning a car if I could access a vehicle for a comparable or lower cost in other ways when I really need it i.e. Car hire/sharing
Q17.4	If fast, convenient, affordable public transport and shared taxis were available to me, I would use private cars/my car less.
Q17.5	I would rather carpool or share a taxi/uber with other travellers for an affordable cost than take public transport.
Q18.2	I use transport planning websites and apps (e.g. Google Maps or TransLink Journey Planner) regularly.
Q18.3	I find it easy to plan my journeys.
Q18.4	Information about public transport is easy to find.
Q18.5	I would be happy to use my phone to tap on and off and pay for transport services.
Q18.6	I would be happy to use my debit card to tap on and off and pay for transport services.
Q19.2	I like the idea of having flexible access to my choice of car with included insurance, maintenance and roadside assistance for a weekly or monthly fee.
Q19.3	It would be useful to have access to a car on a month by month basis.
Q19.4	I would be willing to pay between $400 and $600 a month for a basic car subscription with all vehicle costs covered (excluding fuel).
Q19.5	I would be willing to pay between $600 and $1200 a month for a luxury car subscription with all vehicle costs covered (excluding fuel).
Q19.6	It would be good to be able to reduce my expenses by only paying for a vehicle when I need one.
Q19.7	I would consider using a car subscription service instead of buying or leasing a new car myself.
Q20.2	I like the idea of having all my transport costs bundled into a convenient monthly subscription package or bill – like a mobile phone or internet plan.
Q20.3	I would want my 'transport credits' in a mobility package to rollover to the next month if they were not used.
Q20.4	I would prefer that money was only deducted from my travel account when I complete a trip, rather than having a pre-paid fixed monthly cost for all my regular trips.
Q20.5	If my transport package included other modes I haven't used before, I would try them.
Q20.6	I would give up car ownership or choose not to buy a car if affordable, fast, and convenient public transport, active travel, or taxi style options were available to me.
Q20.7	I would be likely to use Mobility as a Service for regular trips, but I would still want to own a car.
Q21.2	I support electric vehicles because they are better for the environment.

Q21.3	Electric vehicles have enough battery range (kilometres) to get me to and from the places I need to go.
Q21.4	I would prefer a hybrid vehicle over a petrol or diesel vehicle.
Q21.5	I would happily charge an electric vehicle at home or work using a normal powerpoint or electric vehicle charger.
Q21.6	For longer trips I would happily charge my electric vehicle for 20-30 minutes every 2-4 hours at fast chargers.
Q21.7	Electric vehicles are too expensive to buy even if the running and maintenance costs are lower.
Q21.8	I would consider buying an electric vehicle for my next car.
Q21.9	I would buy an electric vehicle if there were cost discounts, subsidies, or rebates.
Q22.2	I like the idea of safe automated vehicles if they make it easier for me to travel.
Q22.3	I trust automated vehicles to operate safely.
Q22.4	I would consider buying an automated vehicle.
Q22.5	I would be comfortable riding in an automated vehicle.
Q22.6	I would allow my children to be transported in an automated vehicle.
Q22.7	I would pay a higher price for a partly or fully automated vehicle than a non-automated vehicle.
Q22.8	I would enjoy being able to use my time in an automated vehicle for things like reading, or relaxing rather than driving.
Q22.9	I enjoy actually driving a car, and would prefer that to riding in an automated vehicle.

Appendix B – Code

All code used for generating the figures and analysis in this report was uploaded to GitHub under an MIT licence:

https://github.com/lachlan-git/predict_sp

Appendix C – Confusion Matrices

To analyse the specific performance of the model, confusion matrices where generated for certain questions. These give a broad Idea of the accuracy of the models suggested in this report. Q19.5 is a typical training set question, Q22.2 is well performing test set model, and Q22.3 is a poor performant test question.

Naïve Model

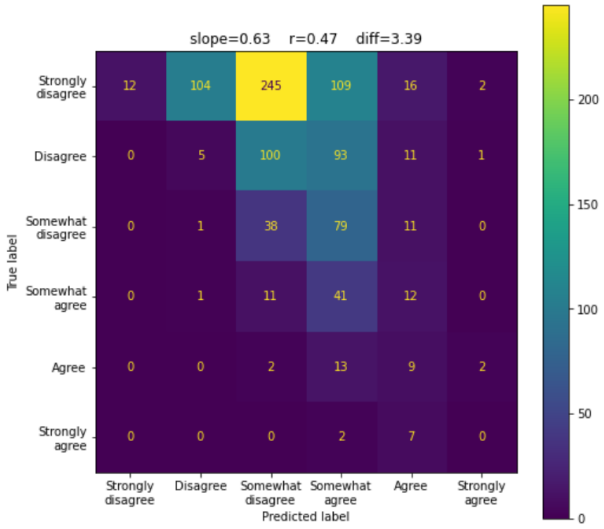


Figure 33 - Confusion Matrix for Q19.5 (training set)

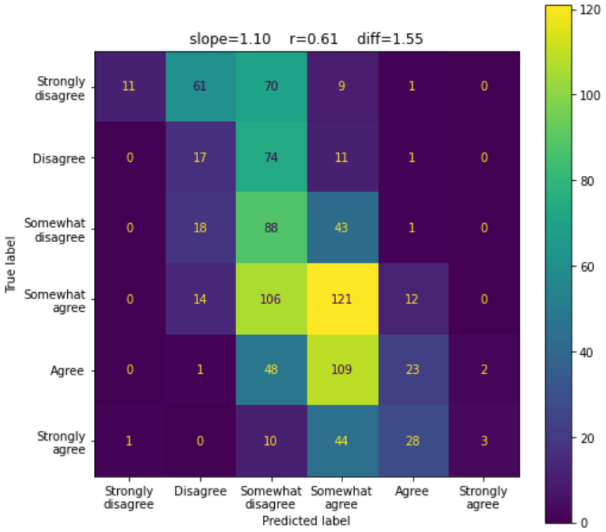


Figure 34 - Confusion Matrix for Q22.2 (test set)

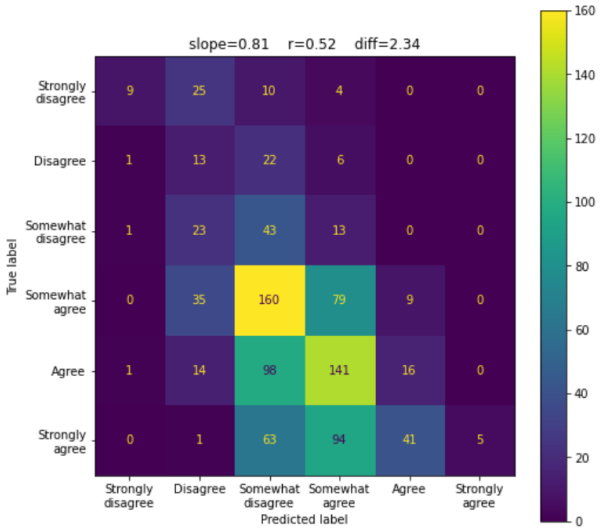


Figure 35 - Confusion Matrix for Q22.3 (test set)

No Compression

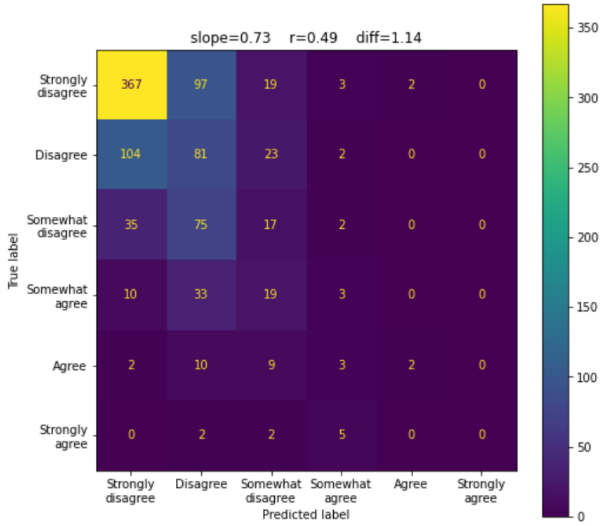


Figure 36 - Confusion Matrix for Q19.5 (training set)

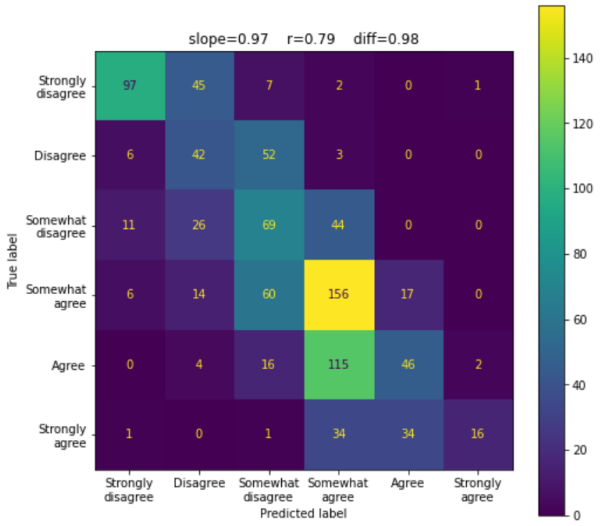


Figure 37 - Confusion Matrix for Q22.2 (test set)

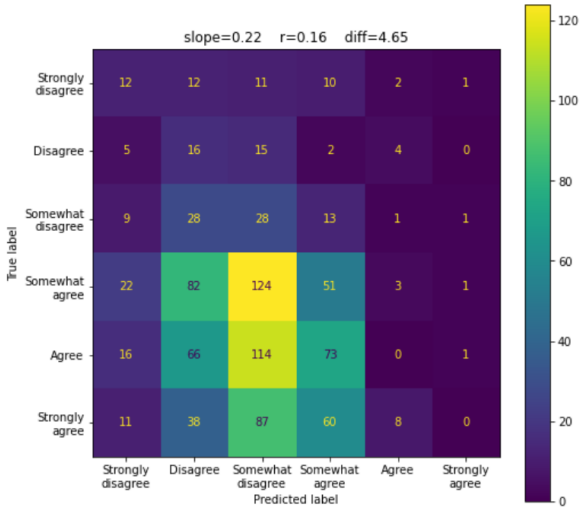


Figure 38 - Confusion Matrix for Q22.3 (test set)

PCA Results 2 dimensions of Heterogeneity

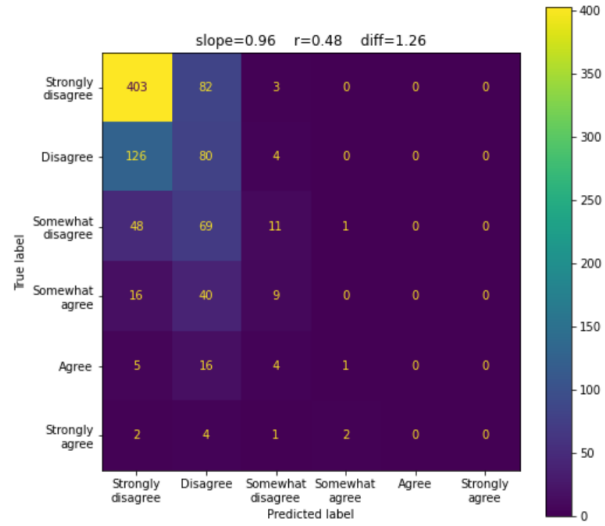


Figure 39 - Confusion Matrix for Q19.5 (training set)

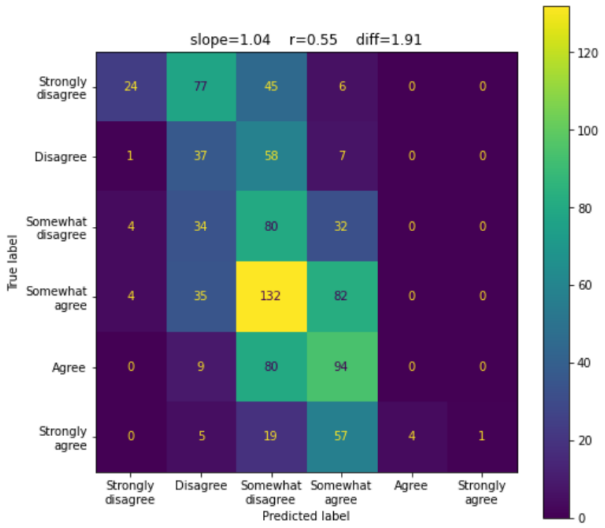


Figure 40 - Confusion Matrix for Q22.2 (test set)

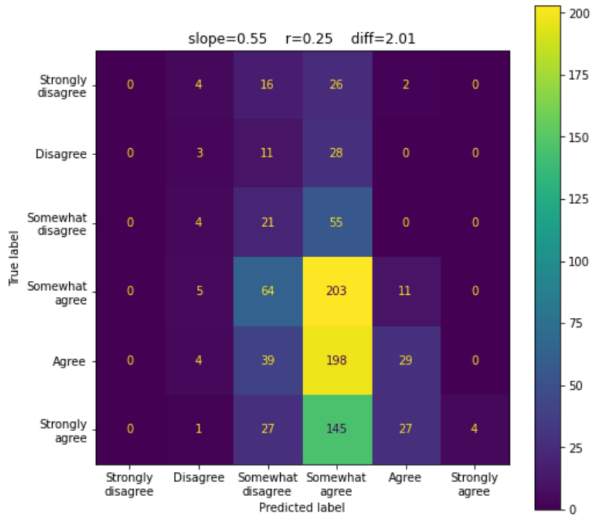


Figure 41 - Confusion Matrix for Q22.3 (test set)

PCA Results 5 dimensions of Heterogeneity

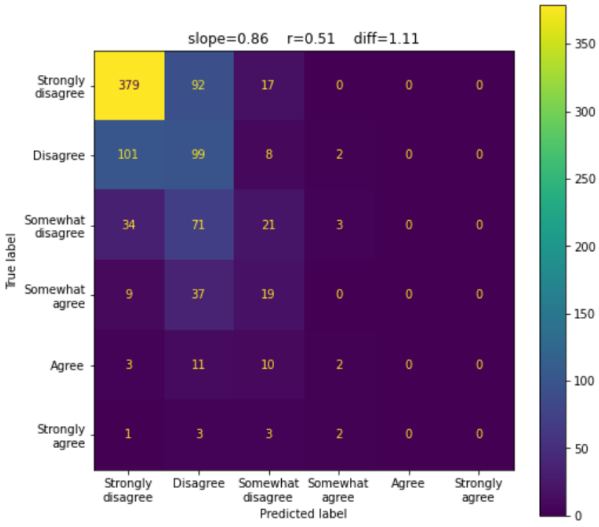


Figure 42 - Confusion Matrix for Q19.5 (training set)

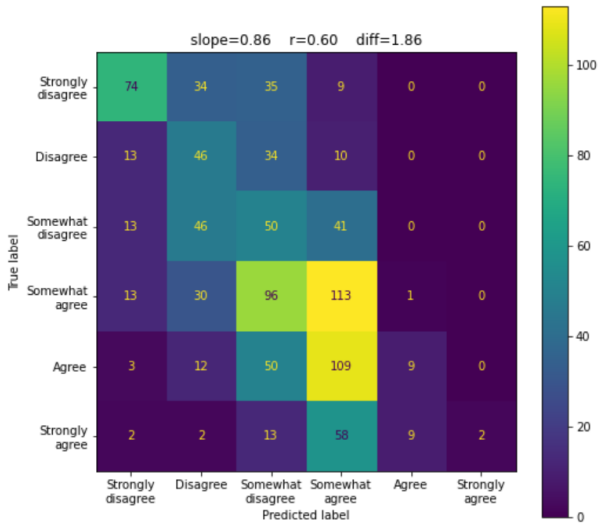


Figure 43 - Confusion Matrix for Q22.2 (test set)

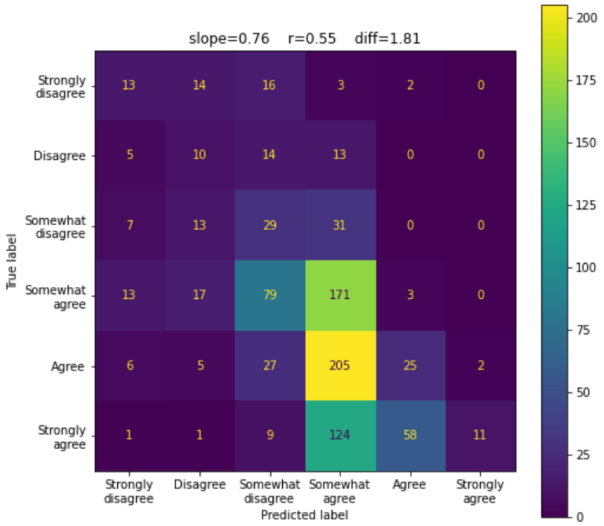


Figure 44 - Confusion Matrix for Q22.3 (test set)

Auto Encoder

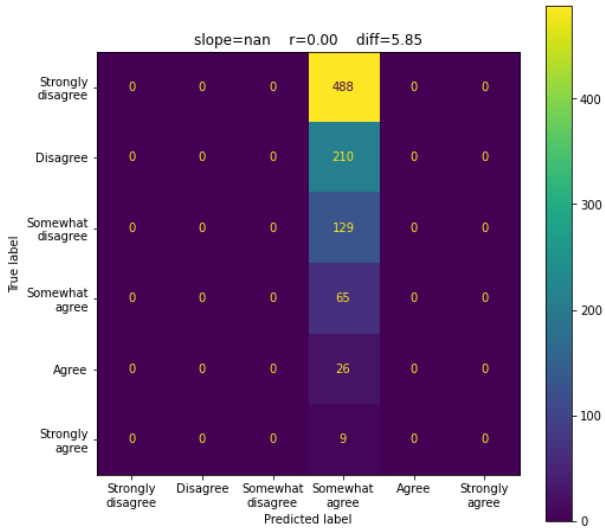


Figure 45 - Confusion Matrix for Q19.5 (training set)

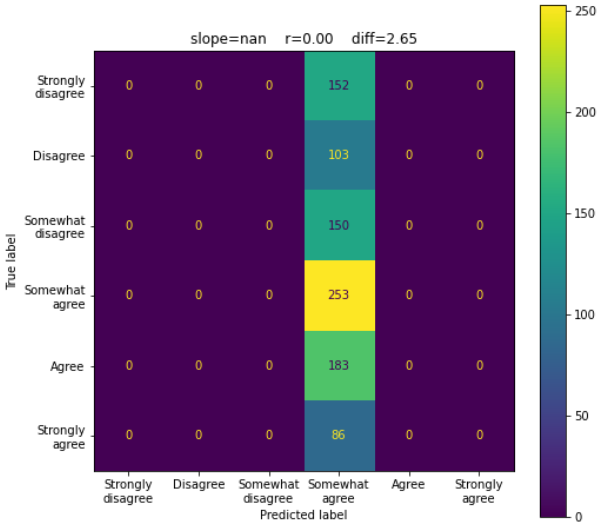


Figure 46 - Confusion Matrix for Q22.2 (test set)

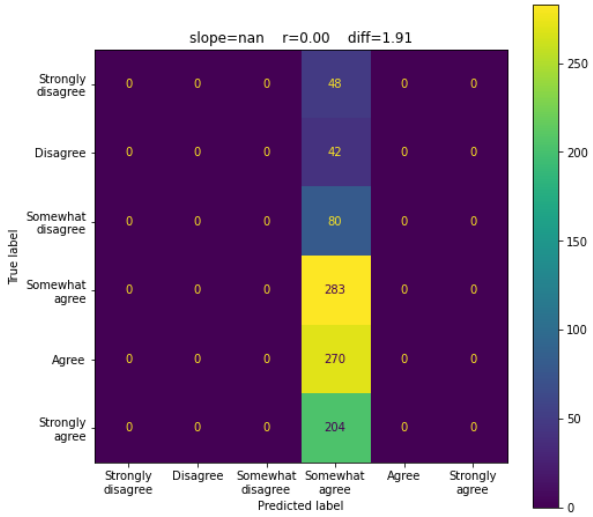


Figure 47 - Confusion Matrix for Q22.3 (test set)

Sparse Auto Encoder

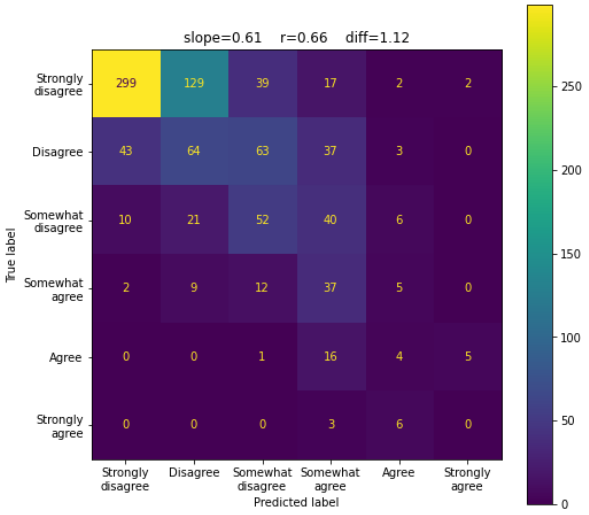


Figure 48 - Confusion Matrix for Q19.5 (training set)

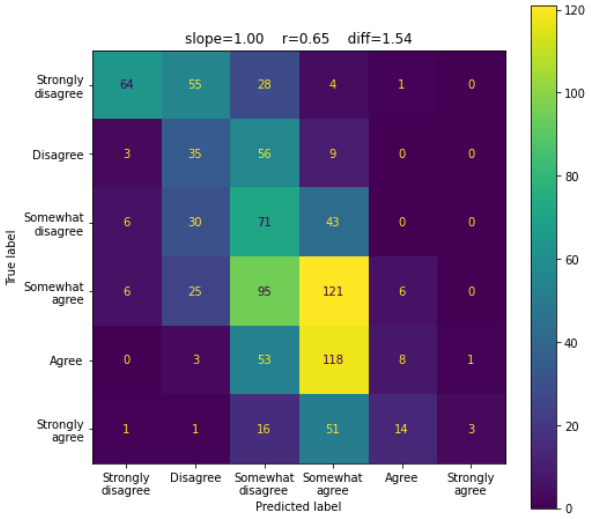


Figure 49 - Confusion Matrix for Q22.2 (test set)

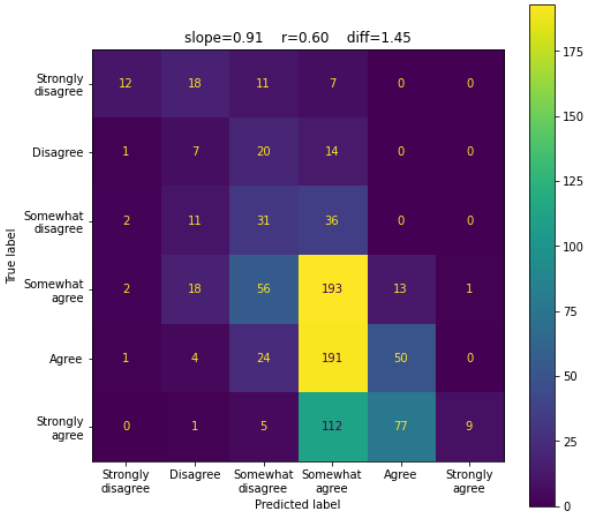


Figure 50 - Confusion Matrix for Q22.3 (test set)

Premeasure results

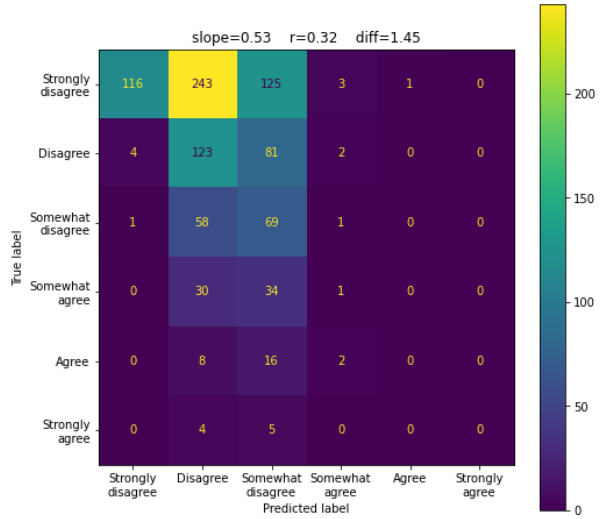


Figure 51 - Confusion Matrix for Q19.5 (training set)

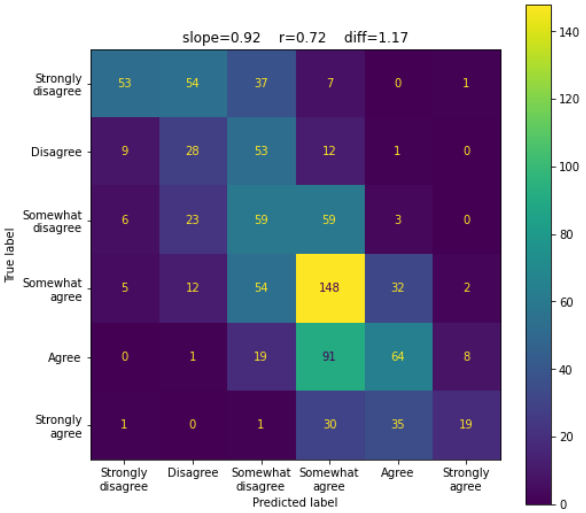


Figure 52 - Confusion Matrix for Q22.2 (test set)

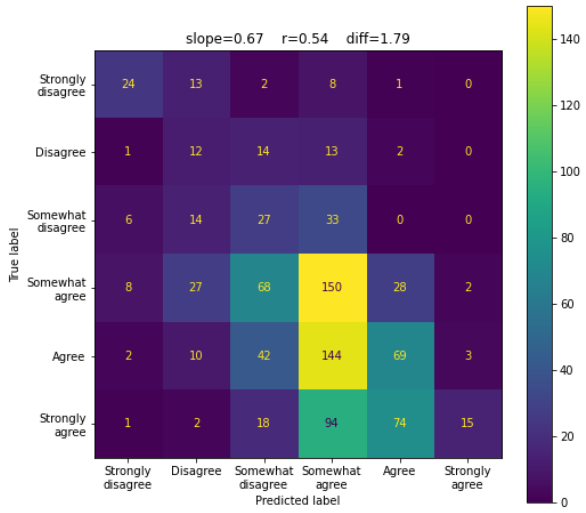


Figure 53 - Confusion Matrix for Q22.3 (test set)