Course: Data Analytics Major Project and Placement MOD007894

Module Code: MOD007894 TRI3 F01CAM

Assessment Element 011 – Dissertation

Student ID: 2063656

Date: 15/09/2022

Project title:

Obstinate customer support clients: detection, analysis, and time series modelling using machine learning

Declaration page

I, Maxim Khovanskiy, hereby declare that the work in this dissertation, titled "Obstinate customer support clients: detection, analysis, and time series modelling using machine learning", has been carried out by me in fulfilment of the requirements for the award of the degree of MSc Applied Data Science (Conversion) and to Anglia Ruskin University; it has not been previously submitted to this or any other educational institution for the award of a degree or educational qualification. I also declare that the information published in this dissertation has been obtained and presented in accordance with the academic rules and ethical conduct protocol. Any information obtained from other sources has been properly referenced.

Acknowledgement page

I would like to express my gratitude to my academic supervisor, Raj Shukla, and my personal tutor, Mahdi Maktab-Dar-Oghaz, for overseeing the project. Additionally, I would like to give a special thanks to my workplace supervisor, Alison Hartshorn, for her collaborative efforts in identifying interesting areas of research, as well as for supplying the data needed to carry out said research.

Finally, I would like to thank my parents, whose support was crucial to the completion of this project.

Table of Contents

List of figures

List of tables

List of equations

$$Z = \frac{\sqrt{n}(x - \mu)}{\sigma}$$

Abstract

Customer support staff at Anglia Ruskin University noted an interesting phenomenon: in LiveChat, an interface that allows real-time communication between university staff and prospective students, a small number of clients (students) demonstrated a tendency of making a large number of trivial or previously fulfilled requests, often over the course of many independently initiated LiveChats. Thus, despite making up only a small percentage of all clients, these obstinate clients would reportedly take up a significant proportion of the customer support staff' time. Previous research has largely focused on detecting and analysing cases of related but distinct phenomena such as harassment and spamming; the subject of this project differs from these phenomena in that it doesn't appear to involve malicious intent, and as such requires a separate investigation. In this project, I use anonymous data from 9 months' worth of Anglia Ruskin's LiveChat logs to 1) formulate a plausible and robust definition of obstinate clients, 2) build models that

detect obstinate clients a) from the staff's responses and b) given the available metadata, 3) examine the obstinate clients' effect on the staff's resources, and 4) build a time series model for the prevalence of obstinate clients per month. My findings indicate that, despite making up only 4% of all clients, obstinate clients contribute 33% of the total text volume*, initiate 16% of the LiveChats, and take up 16% of staff time. They also demonstrate that it is possible to detect, with limited reliability, obstinate clients from metadata, suggesting that prophylactic measures can be taken. A similar degree of reliability in the detection of obstinate clients has been demonstrated with staff responses, potentially suggesting that the staff's interaction with clients can be optimised to minimise obstinate behaviour; however, the direction of causality between staff responses and obstinate behaviour by the client has not been examined in this project, so the question of whether or not the former has a significant influence on the latter remains open. Additionally, time series modelling predicts strong seasonality in obstinate client prevalence, potentially implying high predictability and hence suggesting that pre-emptive measures may be effective, although more evidence is needed for a conclusive verdict on this hypothesis.

** "Volume of text", in the context of this paper, refers to the total length of characters contained within a piece, or aggregation, of text.*

Chapter 1: Introduction

1.1 Overview

Since LiveChat functionality was implemented into Anglia Ruskin University's website in September 2021, several members of customer support staff reported that the efficiency of their services was being compromised by a minority of obstinate students who would make a number of requests that could easily be fulfilled without customer support (such as inquiring about the start date of a course, information on which is readily available on the university's website), make the same request repeatedly even when it was already fulfilled (such as repeatedly demanding further information when it had been made clear that no relevant information is available), and keep repeating the process over many different LiveChat messages and independent LiveChats. Since these students would send disproportionately large

quantities of text, including making a large number of independent requests, a disproportionate amount of time and effort was reported to have been spent dealing with them, leaving less time and effort for the rest of the students. Moreover, due to the trivial and repetitive nature of the requests being made, many members of staff believed the LiveChats involving these obstinate students to have been unproductive, thus reducing the total productivity-to-time-spent ratio of the service they offered.

Therefore, there is currently a high demand within the university for tools that would help handle obstinate students. Such tools can mainly be categorised as follows:

1. Guidance tools, which instruct the staff on, or assist them with, interacting with obstinate students
2. Detection tools, which inform the staff on how likely their client is, or whether their client is likely, to engage in obstinate behaviour
3. Group prediction tools, which inform the staff on the predicted proportion or quantity of obstinate clients
4. Awareness tools, which provide useful information on the data that has been collected so far about obstinate students

The aim of this project is to lay the groundwork for the devisal of tools in each of these four categories. For a more detailed description of the research aim, refer to *1.3 Research Aim.*

Additional information provided by the customer support staff includes the observation that obstinate students appear to have similar tendencies in writing, such as the preference for particular words and phrases. This claim will be investigated as part of the process towards completing one of the principal objectives, to be outlined in *1.4 Research Objectives*.

1.2 Problem Background

One of the main rationales behind the project is that the phenomenon of obstinate customer support clients (to be henceforth referred to as the *OCSCP)* is, for the most part, not documented in known literature. Previous research on the topic of electronic communication analysis has primarily focused on other phenomena such as

harassment, spamming, phishing, or sentiment analysis. Note that this paper uses the term "obstinate clients” without implying any negative connotations. In fact, as found in [R. Mano, G. Mesch, 2012, "E-mail and work performance"], the number of emails received and sent positively correlates with work performance; it’s not implausible that the same is true for LiveChat messages.

Admittedly, since both the OCSCP and these phenomena fall under the same umbrella of electronic communication analysis, a number of effective techniques relevant to the OCSCP have been developed. For example, THEMIS [I. Fette, N. Sade, A. Tomasic, 2007, "Learning to Detect Phishing Emails"], which uses deep learning to detect phishing from the contents of emails, and PILFERS [Y. Fang, C. Zhang, C. Huang, L. Liu and Y. Yang, 2019, "Phishing Email Detection Using Improved RCNN Model With Multilevel Vectors and Attention Mechanism"], which detects phishing from factors that don’t utilise text analysis (such as sender URL and the presence of JavaScript) can both be repurposed for the problem of the OCSCP or be used as the basis for new, custom-developed techniques. However, for the very reason that the literature on electronic communication analysis techniques is already plentiful, the development of novel effective techniques will not be the focus of this project.

Instead, this project will focus on laying a comprehensive groundwork for further research into a mostly undocumented phenomenon, and in doing so will prioritise breadth, thoroughness, and fundamental soundness over effectiveness and immediate usability; parts of it may be thought of as proofs-of-concept. More effective techniques and consumer-ready tools can be developed in follow-up projects.

In summary, since the OCSCP isn’t well-documented in existing literature, there exists a literary gap between the available electronic communication analysis techniques and their application to the specific phenomenon of the OCSCP, which

this project attempts to fill. The market justification for attempting to fill this gap has been provided in *1.1 Overview.*

To this end, I formulate the following fundamental research hypotheses:

1. It is possible to reliably detect obstinate clients from the available metadata about the client.
2. The staff's interaction with a client influences the probability that said client will engage in obstinate behaviour, and this influence can be identified using the available data.
3. Obstinate clients take up a statistically significant proportion of the customer support staff time.
4. There is a clear and predictable pattern in the prevalence of obstinate clients throughout the course of the year.

1.3 Research Aim

The overarching goal of the project is to establish key results that will enable follow-up projects to build each of the four categories of tools characterised in *1.1 Overview*, as well as to develop methods using which further research into the OCSCP can be conducted.

This will be done by testing each of the research hypotheses outlined in *1.2 Problem Background*; all the relevant methods and observations will be described during the process of testing these hypotheses, thus completing the second half of the research aim.

Since the available data does not include labels of obstinate clients, as well as since the aim is to make my finding conducive to further research, obstinate clients will first need to be defined appropriately before the hypothesis testing phase.

1.4 Research Objectives

The following objectives constitute a compact summary of this project's research protocol; as such, the project's research methodology will be built around these objectives, and they will consequently be referenced all throughout the paper:

Objective 1: Formulate the definition of obstinate clients in such a way that it is both likely to encapsulate all of the behaviours described in the first paragraph of *1.1 Overview* (is plausible) and can be applied automatically to unclassified clients (is robust).

Objective 2: Train a classification model which can reliably detect obstinate clients from staff messages.

Objective 3: Train a classification model which can reliably detect obstinate clients from metadata about the client.

Objective 4: Compute the percentage of clients who are obstinate, the total percentage of staff time taken up by obstinate clients, the total percentage of LiveChats initiated by obstinate clients, and the percentage of the total volume of text sent by obstinate clients; investigate whether the former of these statistics is statistically significant.

Objective 5: Build a time series model that forecasts the total traffic of obstinate clients per month.

1.5 Research Scope

The project will utilise the following statistical methods:

- a range of machine learning techniques, including:
    - classification and regression algorithms, such as multi-layer perceptrons and support-vector regression
    - hyperparameter grid search
    - bag-of-words vectorisation

    - statistical modelling via classification and regression algorithms and via time series modelling ─ specifically, *ARIMA* modelling ([R. H. Shumway, D. S. Stoffer, 2017, "Time Series Analysis and Its Applications: With R Examples"])
    - the Augmented Dickey-Fuller test of stationarity
- several plotting methods, such as:
    - Regular line graph
    - Multiple line graph
    - Scatter plot
    - Autocorrelation function plotting
    - Partial autocorrelation function plotting

The project will take into account the following parameters:

- 30 pieces of metadata, including location, device used, most recently visited page, and others, about each of the 2011 clients
- The LiveChat transcripts for each LiveChat initiated by each of the 2011 clients
- Securely tokenised student IDs of each of the 2011 clients
- A number of custom-built parameters such as names of each the members of staff involved in each of the LiveChats (data about staff names in the original datasets are inaccurate), the tokenised student IDs of all the clients classified as obstinate, the number of obstinate clients in each of the 9 months (except April, for which data was missing) for which data is available, and others

The project will measure the following significant quantities:

- The balanced accuracy and sensitivity of the classification models described in *Objective 2*
- The quantities described in *Objective 4,* as well as the z-score for the total percentage of staff time taken up by obstinate clients and the percentage of clients who are obstinate
- The monthly total number of obstinate clients and corresponding LiveChats initiated, the percentage of clients who are obstinate and the percentage of LiveChats that are initiated by obstinate clients, and the total number of clients

and LiveChats for each of the 9 months (except April, for which data was missing).

The project will not include:

- Comparative analysis to find the most effective classification models due to hardware limitations
- Hyper-parameter grid search for all of the classification models due to hardware limitations
- Tests of whether the distribution of quantities that are assumed to be normally distributed (such as the amount of staff time a client takes up) are actually normally distributed, as it is left for follow-up research
- Classification and time series models that are consumer-ready due to insufficient data.
- Tests of statistical significance for the statistics described in *Objective 4* unless otherwise specified in *Objective 4* because they are not directly related to customer support staff's description of obstinate students.
- Causality analysis for a definitive verdict on research hypothesis 2 (as described in *1.2 Problem Background.*

1.6 Methodology

After a series of data collection and data-cleaning procedures, obstinate clients will be preliminarily defined as those who have sent a statistically outlying cumulative volume of text. This definition guarantees that those classified as obstinate will have sent a disproportionate total volume of text and also makes it likely that they will have taken up a disproportionate amount of staff time. Subsequently, a neural network will be trained on two cross-validated samples independently, one containing resampled balanced data and the other containing original imbalanced data. The better-performing neural network will be selected. The rationale behind this step is that, if there are common patterns in the obstinate clients' live chats, such as making repetitive and/or even trivial requests, the network will be expected to pick up on them. Finally, obstinate clients will be redefined as those classified as such by both the preliminary definition and the neural network; this combines the benefits of

both and thus makes it likely that all of the reported obstinate behaviours are encapsulated.

Subsequently, two neural networks will be trained: one to detect obstinate clients from staff message data and the other to detect them from metadata. The former is intended as laying a foundation for future research, and hence won't be hyperparameter-grid-search for improved performance; the latter, intended to be mostly consumer-ready, will be hyperparameter-grid-searched. The performance of both models will be assessed.

The quantities described in *Objective 4* will be computed. Then, the amount of staff time taken up by each student will be assumed to be normally distributed, and a hypothesis test for statistical significance at the 0.05 level of the average value for obstinate students will be run.

Finally, using ACF and PACF plots, an appropriately parameterised ARIMA model will be fitted to the data containing the prevalence of obstinate clients every month. The model will be used to forecast the prevalence of obstinate clients for the following months, and any observed patterns will be noted and explained. Furthermore, an SVR model will also be fitted, which could prove to be effective if the underlying pattern is strongly seasonal.

1.7 Contribution

The most significant finding of the project is that obstinate client prevalence per month demonstrates a clear seasonal pattern; while the evidence is inconclusive as to whether this trend is an anomaly or a representation of a larger global trend, these preliminary results can already be experimentally deployed by customer support staff at Anglia University to prepare for influxes of obstinate clients in advance.

Moreover, a moderate correlation ($r = 0.4–0.55$) was discovered between the total number of LiveChats initiated per month and the percentage of these LiveChats that were initiated by obstinate clients. The underlying cause of this correlation has not been examined, and constitutes another possible area of future research.

Additionally, the project has shown that it is possible to detect obstinate clients from metadata with limited reliability; while the detection model is imperfect, it can also already be used to assist customer support staff at work. Moreover, the methods used in the project can be applied to greater volumes of data in follow-up projects, which would likely improve the performance of the model.

Finally, the hypothesis that obstinate students take up a statistically significant amount of staff time has been proven true under some general assumptions, although the question of the validity of these assumptions is left for follow-up investigations to explore.

Chapter 2: Literature Review

2.1 Overview

The OCSCP is the reported phenomenon of a minority of customer support clients making a disproportionately large number of requests ─ often trivial or previously addressed requests ─ and often on a large number of independent occasions. This behaviour, referred to as “obstinate” in this paper, is distinct from related behaviours such as harassment and spamming in that it bears no malicious intent ─ it has been reported by staff that the apparent objective of such clients is simply to receive the information requested.

This phenomenon has not been studied previously, and there are few to no accounts of it other than that provided by the International Marketing & Communications department of Anglia Ruskin University. Consequently, no research into this

phenomenon has been published to date. However, research into a number of related domains has been carried out, and some of the methods employed, and results derived, by this class of research will be used in this project. It is this type of research that will be the focus of this chapter; whether or not the methods employed and/or derived results are applicable to the subject of this project will be discussed.

2.2 Electronic Communication Analysis summary

Electronic communication analysis is an umbrella term used in this paper to refer to all domains in which quanta of electronic communication, as defined by the ECPA [Lamba, Manika & Margam, Madhusudhan, 2022, “Sentiment Analysis”], such as messages, user-generated content, emails, etc are analysed in combination with information about the users that produced said quantum of electronic communication.

The following sub-domains of electronic communication analysis have been identified:

- Sentiment analysis
- Phishing, spamming, and spoofing detection
- Harassment, trolling, and cyberbullying detection

The literature on the former two sub-domains will be reviewed; the latter sub-domain will not be reviewed as its standard methods mostly involve a combination of the methods used in the other two sub-domains, as per Alsaed & Derar [Alsaed, Zaina & Eleyan, Derar, 2021, “APPROACHES TO CYBERBULLYING DETECTION ON SOCIAL NETWORKS: A SURVEY”].

2.3 Sentiment Analysis

The following is a generalised version of the taxonomy of sentiment analysis approaches as per Rodrigues, R & Camilo-Junior et al [Rodrigues, R & Camilo-Junior et al, 2018, “A Taxonomy for Sentiment Analysis Field”].

2.3.1 Machine-learning-based

Joshi and Itkat [Joshi and Itkat, 2014, “A survey on feature level sentiment analysis”] propose a three-way classification of central machine learning techniques used in sentiment analysis: *Supervised, semi-supervised, and unsupervised*. The classification distinguishes the three general approaches by what kind of data the machine learning algorithm of choice was trained on:

- Supervised machine learning involves training on labelled data
- Semi-supervised machine learning involves training on a small set of labelled data and a larger set of unlabelled data
- Unsupervised machine learning involves training on unlabelled data

The data used in this project is initially unlabelled; however, neither unsupervised nor semi-supervised learning techniques could be applied to the data as the subject of the study is a very specific category of the data that will be defined as per *Objective 1*. Therefore, most of the project will focus on dealing with labelled data, and so only the supervised machine learning technique is relevant to this project.

2.3.2 Lexicon-based

A *lexicon* is defined by Ramires et al [Rodrigues, R & Camilo-Junior et al, 2018, “A Taxonomy for Sentiment Analysis Field”] as a “collection of terms with their respective emotional scores”. Lexicon-based sentiment analysis aggregates the emotional scores of all the known terms in a given text and classifies them into pre-selected sentiment categories. One drawback of lexicon-based approaches is that text containing terms that do not feature in the lexicon will not be analysable without manual tweaks to the method. No lexicon has yet been mapped out in relation to the OCSCP; however, this is one potential field of future research (see *5.1 Next steps* for more).

2.3.3 Concept-based

Ramires et al don’t explicitly state their classification criteria for the Concept-based category. However, it is implicit from their work [Rodrigues, R & Camilo-Junior et al,

2018, “A Taxonomy for Sentiment Analysis Field”] that approaches of this category combine the lexicon-based approach with additional contextual information about the available text, such as the sentiment of the product reviewed in case of product reviews, lexical similarity between words for additional insight into their sentiments, etc. Therefore, depending on the problem investigated, it may constitute an improvement over the raw lexicon-based approach, but still suffers from the same fundamental problems. Like with *2.1.2 Lexicon-based*, this type of approach could potentially be tested in relation to the subject of this project in follow-up projects.

2.4 Phishing, spamming, and spoofing

The following is a generalised version of the taxonomy of what N. Q. Do et al [N. Q. Do et al, 2022, "Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions] refer to as *software detection* methods. They don’t explicitly define software detection; however, their taxonomy is applicable to all of spamming, spoofing, and phishing.

2.4.1 List-based

List-based methods use dictionaries of websites and their classification into legitimate and/or suspicious websites to detect quanta of electronic communication containing suspicious links. This method is roughly analogous to the lexicon-based approach in sentiment analysis detailed in *2.1.2 Lexicon-based*, but suffers from additional problems such as the need for the list to be constantly updated to be accurate. This method is not applicable to the subject of the project.

2.4.2 Heuristic-based

This class of method is a derivative of list-based methods and uses data extracted from the list of websites to inform their classification via a manually constructed heuristic. This constitutes an improvement to the raw list-based method as it allows for the websites to be classified automatically, without the need for constant manual updates. Like list-based methods, heuristic-based methods are not applicable to the subject of this project.

2.4.3 Visual-similarity-based

In visual-similarity methods, the visual structure of websites is taken into account when classifying websites into legitimate and suspicious. This class of method can be used in combination with heuristic-based and/or list-based methods, and can improve both the accuracy and the scope of classification. Applying visual machine learning algorithms, such as convolutional neural networks, to the text sent by obstinate clients in the OCSCP is an interesting approach, and may theoretically help detect writing style tendencies, although it's unclear how effective such an approach would be.

2.4.4 Machine learning

Machine learning in relation to phishing, spamming, and spoofing detection can be used to help the classification of websites into legitimate and suspicious, and can hence be used in tandem with any combination of the afore-described methods. It often constitutes an improvement to the heuristic-based class of methods as it can often encapsulate both a higher complexity and a greater diversity of factors. Machine learning will be utilised in this project.

2.4.5 Deep learning

Deep learning is a subset of machine learning that uses a high number of hidden layers, and can help encapsulate an even greater degree of complexity of the factors used in classification, thus improving classification accuracy. This class of methods is applicable to the subject of the project, but will not be utilised due to hardware limitations.

2.4.6 Hybrid

Hybrid methods use a combination of the afore-described methods to achieve greater classification accuracy.

2.5 Taxonomy diagram

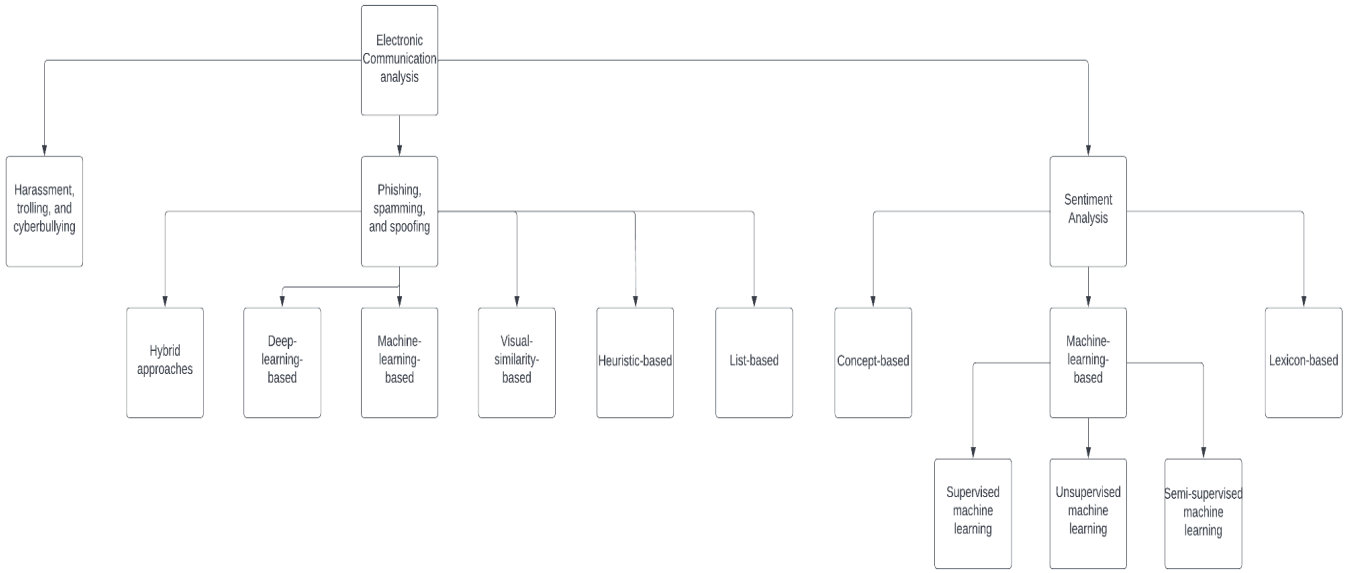


Figure 1 – Taxonomy diagram

Chapter 3: Research Methodology

3.1 Overview

Customer support staff at Anglia Ruskin University reported that a minority of students – referred to as “obstinate clients” in this paper – were consuming significant proportions of their time, often by making trivial requests, initiating many different LiveChats, and sending large volumes of data. Due to a lack of literature on the subject, the staff currently have no knowledge of how to handle these students in

a way that doesn't compromise the amount of support received by the rest of the students. The goal of the project is to provide a methodological basis for how statistical methods can be used to help handle these students, as well as to provide early ready-to-use tools for customer support staff to utilise at their workplace.

The solution concept is as follows:

First, a definition of "obstinate clients" that both closely matches the one provided by ARU customer support staff and can be applied automatically to classify clients (students) en masse is to be formulated. This is to be achieved by combining a heuristic based on the total text volume sent by each client with a neural network trained to guess the heuristic classifications, and using this combination to define obstinate clients. The expected outcome is that the neural network will detect text content patterns of true obstinate clients, while the heuristic will detect text volume patterns of true obstinate clients, and their combination will account for both.

Subsequently, key statistics about obstinate clients will be collected; hypothesis testing will be carried out on relevant statistics, as per *Objective 4*, and training models to detect obstinate clients based on staff messages and the client's metadata, respectively, will be trained and tested.

Finally, several metrics of monthly obstinate client prevalence will be measured and plotted against the total client and LiveChat traffic, and any potential patterns will be noted. These measurements will be used to train two time series models of obstinate client prevalence per month, which in turn will be used to forecast obstinate client prevalence for the near future. These forecasts will be plotted, and any potential patterns will be noted.

All of the procedures described in the latter two paragraphs supervene on the procedure described in the first paragraph; therefore, the procedure described in the first paragraph is crucial for the success of the project.

3.2 Research Framework

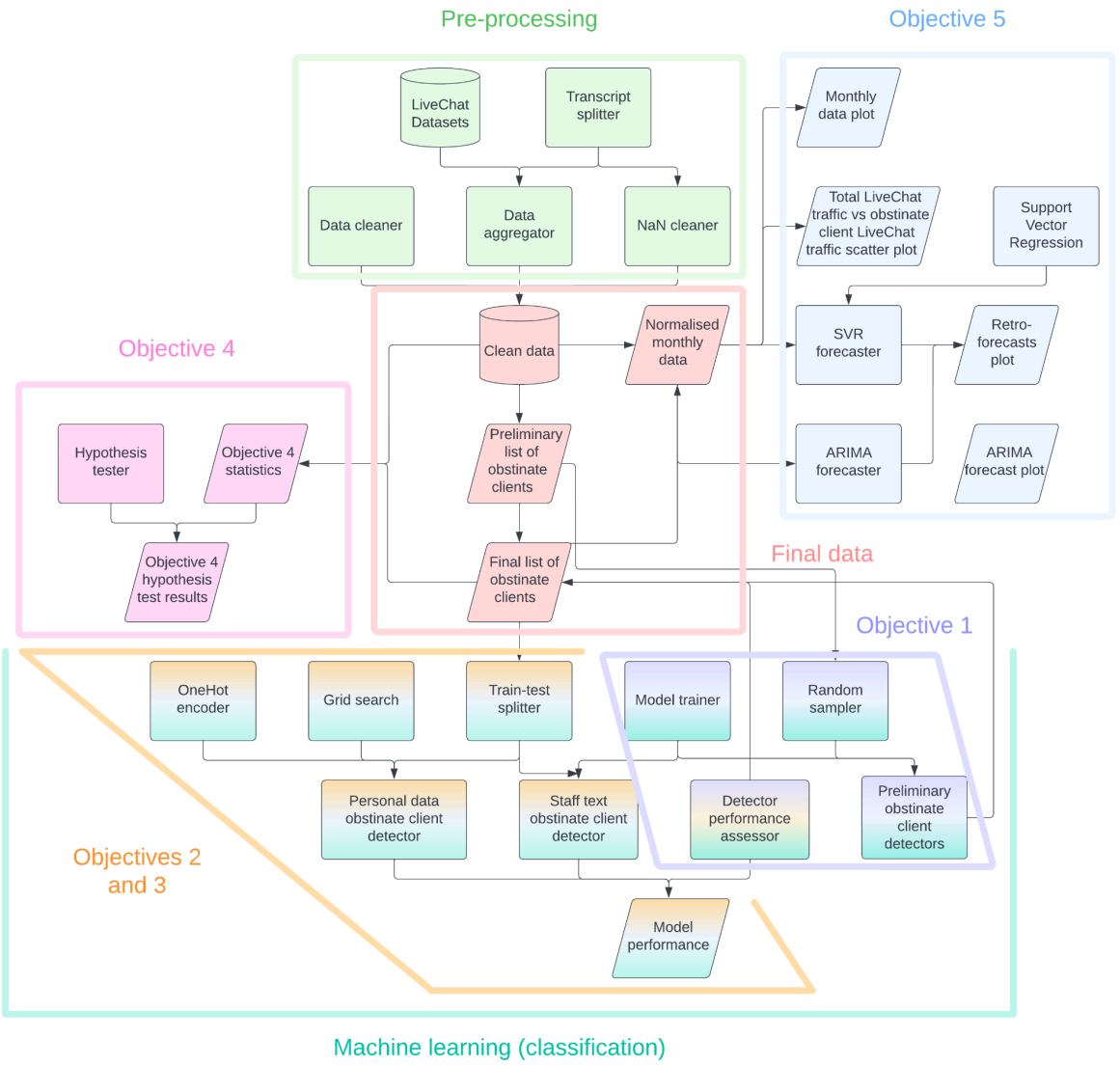


Figure 2 – Data Pipeline

Step 1: Pre-processing.

As a preliminary step, the 9 datasets will be concatenated into one. Subsequently, the research procedure will be initiated.

1) *Transcript splitter*. This module will take the dataset and row index within the dataset as input, and will consist of several sub-components:

a) *Staff name retriever*. This function will, within the transcript, look for the phrase “You are now chatting with”, which is an automatic message sent every time a new member of staff enters the LiveChat and is followed by that member’s registered name. The function will then split all the subsequent text by the space character and retrieve the first element.

b) *Staff and client message retriever*. This function will look for timestamps within the transcript by splitting the transcript into lines, then splitting the lines by the space character, and ultimately looking for those first elements of lines which contain two semicolons (“:”). The timestamps are generated automatically by the system at the start of each line and come in the hours:minutes:seconds format, so this method is guaranteed to retrieve all of them.

   The function then checks if the first element immediately following the timestamp contains a colon.

   - If it doesn’t, it classifies the line as a client message. The system automatically prints the name of the sender of each message, always displayed on a single line, followed by a colon; however, the names of clients were removed from the dataset for privacy protection reasons, so this method is guaranteed to retrieve all, and only, client messages.
   - If it does, it calls the staff name retriever to check if this element, without the colon, is the name of any of the staff; if it is, it classifies the line as a staff message. For the reason described above, this method is guaranteed to retrieve all, and only, staff messages.

   The function then returns the merger of all the messages classified as client messages and, as a separate element, the merger of all the messages classified as staff messages.

c) *Data cleaner*. This is an umbrella designation encapsulating all the data-cleaning methods used throughout the module. These are:

- An array of empty lists will be returned if the transcripts have NaN values.
- In case the first element immediately following the timestamp is a singular colon ─ which is a common artefact of the dataset ─ the third and fourth elements will be instead used. Since the artefact is the result of the afore-described procedure of client names being removed, it only affects client messages, and hence the fact that more than one element is used won't affect the classification.
- In case any of the staff names are of length 2 or shorter, the module will return an array of empty lists. This is a measure to be taken against an artefact of the dataset whereby the entire transcript is corrupted, and hence unusable.
- In case any of the lines are of length 1 or shorter, they will be removed. This is a measure to be taken against individual lines that are corrupted, but whereby the rest of the transcript is intact.
- All formatting code, identified text between a "<" character and a ">" character, will be removed from all messages.

d) *Student ID retriever*. This function will retrieve the tokenised student ID from the dataset. This is an additional piece of information necessary to identify the transcripts with the clients that initiated the LiveChats of these transcripts.

e) *Output*. The module returns the outputs of the staff and client message retriever and the student ID retriever.

2) *Data aggregator*. For every row (LiveChat) in the concatenated dataset, the contents of 30 columns representing the client's anonymous metadata, the output of the transcript splitter applied to the row index and the concatenated dataset, and the cumulative volume of client text retrieved from the transcript splitter will all be allotted to a single object of a class that is to be called *"chat"*; the list of all such objects will be recorded. Additionally, lists of the following will be recorded:

- All client text per LiveChat and its cumulative volume
- All staff text per LiveChat

- All the chats
- All the unique student IDs
- Several technical variables unrelated to the general research methodology structure

3) *Data cleaner*. This module (in this project's implementation, one that is built into the data aggregator module) will implement the following data-cleaning processes:
    - Remove all LiveChats wherein the cumulative volume of all client text is zero ─ this will remove LiveChats which were initiated but never engaged in, and which therefore aren't useful for research.
    - Remove all LiveChats for which the student ID is given as NaN ─ without being able to identify the LiveChat with individual clients, subsequent research methods will not be applicable.
    - Remove all LiveChats wherein the cumulative client text is a duplicate ─ this is a measure against an artefact of the dataset which creates duplicate entries.
4) *NaN cleaner*. This module will handle missing values in the following ways:
    - The mode and state space size of each of the 30 metadata covariates will be recorded.
    - If the state space of a covariate consists of only one state or of too many states (in this project's implementation, more than half the total number of *chats*), and hence conveys zero or close to zero bits of information, that covariate will be removed.
    - For every chat, if the value of a datum of a covariate is missing (is NaN), it will be replaced by the mode for said covariate. An alternative data cleaning strategy would have been to remove all chats containing any missing metadata values; however, it has been found that the implementation of this strategy results in too small of a sample size of chats for any of the subsequent steps to be carried out with a sufficient degree of reliability. Therefore, mean substitution will be implemented instead.

Step 2: Data collection and Objective 1 completion.

1) *Cleaned data*. The data aggregator, data cleaner, and NaN cleaner will all be applied to the concatenated dataset to produce a database containing, together with derivative data, all the information that will be used in the remainder of this project.
2) *Preliminary list of obstinate clients*. Obstinate clients will be preliminarily defined as those whose total text volume over all the LiveChats that they had engaged is a statistical outlier, as per Tukey's definition [Hoaglin, David, 2003, "John W. Tukey and Data Analysis"]. This definition guarantees that those classified as obstinate will have sent a disproportionate total volume of text and also makes it likely that they will have taken up a disproportionate amount of staff time. Since these two are the predominant characteristics of obstinate clients as reported by customer support staff, it is expected that this heuristic would classify most clients the same way that the customer support staff would, with a relatively low false positive rate; however, since it isn't designed to take into account all of the other characteristics described by customer support staff, it doesn't meet *Objective 1*; it is also unlikely to be accurate enough for any consequent conclusions to be drawn with a sufficient degree of confidence. Hence, it will be improved upon in subsequent steps.
3) *Random sampler*. This module will take as input non-obstinate client sample size, dataset, and obstinate client list. It will then perform the following operations:
    - Randomly split the preliminary list of obstinate clients into two groups.
    - Create two mutually exclusive random samples of the specified size of non-obstinate students from the specified dataset.
    - Join each group of obstinate clients with each group of non-obstinate clients.
    - Further split each of the two joint groups into training and testing data.

- Create an array containing information about whether each client in each of the two sets of training data and two sets of testing data is obstinate or not.

Effectively, the random sampler will pre-process the specified dataset for cross-validation, with the additional option of controlling which portion of the dataset is to be cross-validated and under the constraint that the class distribution skew be kept constant ─ which is important for methodological consistency and hence reliability of consequent findings.

4) *Model trainer*. This module will consist of several sub-components:
    a) *Stemmer*. This function will take a string as input and will return a Porter-stemmed [Willett, Peter, 2006, "The Porter stemming algorithm: Then and now"] and data-cleaned version of the string. The data-cleaning involves removing tokens containing "xx" ─ a residue of the anonymisation of student IDs which contains no useful information ─ and an underscore ("_"), which only features in strings (such as _x000D_, which represents a new line) that convey no useful information. For additional functionality, the function will return said version of the string broken down into a list of tokens.
    b) *Vocabulary generator*. This function will take a list of strings as input, apply the stemmer on each of the strings to obtain a list of tokens, and, for each token it had not yet visited, it will record it in a list; it will return the list once it's visited all the tokens that each of the strings in the input list comprises. Thus, the function will output the vocabulary learnt from the given data.
    c) *Vocabulary filter*. This function will strip all the strings in the given data of all tokens that are not part of the output of the vocabulary generator run on said data. This guarantees that any models trained on the data input into the vocabulary generator can still be applied to data containing vocabulary that they weren't trained on.

d) *Bag-of-words vectoriser*. This function will use the bag-of-words method, as per Qader & Ameen [Qader, Wisam & M. Ameen, Musa & Ahmed, Bilal, 2019, “An Overview of Bag of Words;Importance, Implementation, Applications, and Challenges”], as well as the output of the vocabulary generator run on a specified set of data, to convert a list of strings into a sparse matrix. An n-gram range of 1 to 3 will be used in order to account for customer support staff’s reports that obstinate clients tend to prefer not only particular words, but also particular phrases.

e) *Training phase*. A multi-layer perceptron will be trained on the input data, and will then be fitted on the corresponding test data. A multi-layer perceptron was chosen over alternatives such as Naive Bayes because one of the defining characteristics of obstinate clients was reported to be their tendency to make trivial or repetitive requests ─ these appear to require abstract reasoning to detect, which neural networks have been shown to be capable of [David G.T. Barrett, Felix Hill, Adam Santoro, Ari S. Morcos, Timothy Lillicrap, 2018, “Measuring abstract reasoning in neural networks”].

   The module will return the trained model’s classification based on the given training data and the corresponding test data.

5) *Detector performance assessor*. This module will compute the total number of false positives, false negatives, the sensitivity, and the balanced accuracy of a set of predictions compared to a given class distribution.
6) *Final list of obstinate clients*. This model will consist of several steps:
   a) The random sampler will be run on the preliminary list of obstinate clients, the concatenated dataset, and two different non-obstinate client sample sizes: one equal to the sample size of each of the two obstinate client sets and one such that no elements of the cleaned data remain unsampled. Thus, two sets of training data will be fed into the model trainer: one with a

balanced but unrepresentative class distribution and one with an imbalanced but representative class distribution. Consequently, since both sets of data will be split into two by the random sampler, 4 total sets of data will be output.

b) The 4 sets of data will be input into the model trainer.
c) The model trainer's classification will be input into the detector performance assessor, and the performance of the pair of models trained on balanced data and the pair of models trained on imbalanced data will be compared; the higher-performing pair of models will be selected. The performance metrics to be used for this comparison are discussed in *3.4 Evaluation metrics*.
d) Obstinate clients will be redefined as belonging to the subset of the preliminary obstinate clients who weren't classified as obstinate clients by either of the selected pair of models. Since the new definition only includes clients classified as obstinate by both the heuristic classification and the trained neural network, the false positive rate with regard to customer support staff's hypothetical classification should drop significantly. Moreover, all of the behaviours originally described by customer support staff are likely to be captured by the definition: the increased volume of text, number of LiveChats initiated, and amount of staff time taken up per client are expected to be accounted for by the heuristic classification, while behaviours such as trivial and repetitive requests, as well as preference for particular words and phrases, is expected to be learnt by the neural networks. Therefore, this step is expected to complete *Objective 1*.

Step 3: Objectives 2 and 3 completion.

1) *Train-test splitter*. The data will be randomly split into training and testing data, with the former constituting 70% of the input data.
2) *Metadata obstinate client detector*. The train-test splitter will be applied to cumulative staff text, and the model trainer will be fed the output of the train-splitter. Hence, the output of this stage will be the distribution of obstinate and non-obstinate clients predicted by the neural network.

3) *Grid search*. This function will perform a cross-validated hyperparameter grid search for the specified classification algorithm on the given data and over the specified hyperparameter grid. Moreover, the function will give the option to iterate the grid search a chosen number of times; in this case, the hyperparameters that yielded the best individual iteration performance will be selected, and the set of these hyperparameters, as well as their average performance over all the iterations, will be output.
4) *OneHot encoder*. This function will take categorical variable data as input and convert it into a binary matrix via OneHot encoding, as per Kedar & Pardawala [Potdar, Kedar & Pardawala, Taher & Pai, Chinmay, 2017, “A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers”].
5) *Staff text obstinate client detector*. The train-test splitter will be applied to the metadata dataset, retrieved from the cleaned data database; the output will then be fed through the OneHot encoder and the grid search over a number of applicable pre-selected hyperparameters and 3 iterations (number of iterations being limited by the computational strain on the available hardware). Hence, the output of this stage will be the distribution of obstinate and non-obstinate clients predicted by the neural network.
6) *Model performance*. The detector performance assessor will be used to gauge the performance of the two models, and the appropriate conclusions will be drawn. As part of the conclusion-drawing process, the veracity of the claim described in *1.1 Overview* will be evaluated.

Step 3: Objective 4 completion.

1) *Objective 4 statistics*. Using information from the cleaned data database, the following statistics will be computed (in the project’s implementation, one of the statistics was collected from the original concatenated dataset directly):
    - The percentage of all clients who are obstinate
    - The total percentage of staff time taken up by obstinate clients

- The total percentage of LiveChats initiated by obstinate clients
- The percentage of the total volume of text sent by obstinate clients (to be referred to as *PTVOC* from now on*)*

2) *Hypothesis tester*. First, the z-score of the PTVOC will be computed using the following formula [E. Kreyszig, 1979, “Advanced Engineering Mathematics (Fourth ed.)”]:

$$Z = \frac{\sqrt{n}(x - \mu)}{\sigma},$$

where *Z* is the z-score, *n* is the sample size (number of obstinate clients), *x* is the mean staff time taken up per client within the sample (of obstinate clients), μ is the is the mean staff time taken up per client among the parent population (all the clients), and σ is the standard deviation of staff time taken up per client.

Second, the critical value of the standard normal distribution at the chosen level of significance will be retrieved.

Finally, the critical value will be compared to the z-score; if the z-score is higher than the critical value, then the result will be designated as statistically significant given the underlying assumptions.

3) *Objective 4 hypothesis test results*. The hypothesis tester will be run on the PTVOC at the 5% significance level, and the appropriate conclusions will be drawn.

Step 4: Objective 5 completion.

1) *Normalised monthly data*. The same data-cleaning procedure as described for the data aggregator module will be carried out for each of the 9 original, unconcatenated datasets.

Subsequently, the following statistics will be computer for each of the 9 months ─ apart from April, for which the tokenised student ID data is completely missing ─ will be computed:

- Total number of LiveChats initiated
- Total number of clients

- Number of obstinate clients
- Number of LiveChats initiated by obstinate clients
- Percentage of clients that are obstinate
- Percentage of LiveChats that were initiated by obstinate clients

For April, a variation of mean substitution will be performed; each of these statistics will be replaced by the mean value of the same statistic for April's neighbouring months ─ March and May.

Finally, each of these statistics will be normalised in order for all of them to be plottable on the same graph.

2) *Monthly data plot*. The above statistics will all be plotted on the same graph, and the appropriate observations will be made.
3) *Total LiveChat traffic vs obstinate client LiveChat traffic scatter plot*. The total number of LiveChats will be plotted against the percentage of LiveChats that are initiated by obstinate clients to investigate a possible correlation; the appropriate observations will be made. Furthermore, the correlation coefficient will be computer, and the appropriate conclusions will be drawn.
4) *Support Vector Regression*. This function will take as input the normalised monthly data over time and will fit it onto Support Vector Regression with an epsilon value of 0.01 (higher values do not produce accurate results).
5) *SVR forecaster*. The total number of LiveChats initiated per month, as well as the percentage of LiveChats initiated by obstinate clients per month, will be fed into the Support Vector Regression module. Under the assumption that there is strong seasonality in the obstinate client LiveChat time series, this model can be tentatively used to forecast future trends until more data become available.
6) *ARIMA forecaster*. First, an Augmented Dickey-Fuller test will be performed to confirm that the time series is stationary. Given that there are only 9 months of data available, a global trend, even if it exists, isn't expected to be detectable in such a short time frame, so the test is

expected to be passed. If so, an integrated order of 0 will be selected. Otherwise, the first difference will be taken, and the Augmented Dickey-Fuller test will be performed again; if it is passed, an integrated order of 1 will be selected. This process will be reiterated until the test is ultimately passed and the corresponding integrated order will be selected.

Secondly, the ACF and PACF values will be plotted. They will be investigated for Moving Average and Auto-Regressive signatures, and the appropriate MA and AR orders will be selected.

Finally, the monthly obstinate client LiveChat data will be fit onto an ARIMA model with the selected orders. The model will then be able to forecast future trends.

7) *Retro-forecasts plots*. The SVR and ARIMA forecasters will be used to retro-forecast the data that they were trained on, and these forecasts will be plotted alongside one another and overplayed on top of the true obstinate client LiveChat data. This is done to verify that there isn't any significant mismatch between the retro-forecasts and the true data; if there is, that would indicate that the models that significantly mismatch the data are unlikely to be reliable.
8) *ARIMA forecast plot*. Unlike SVR, ARIMA can forecast future trends with comparatively little overfitting. Therefore, while 9 months is likely too small of a sample size for any forecasts to be reliable ─ especially since annual seasonality is very likely ─ tentative forecasts can still be made to observe if there are any trends. If there are, they will be noted.

3.3 Data Collection/Structure

The original datasets contain many missing values as well as numerous values which aren't useful. The exact data-cleaning methods were described in detail in *Research Framework 3.2*. Having performed data-cleaning, the following are the modes of each of the metadata covariates:

MSC	Agent Name	Agent Login Name	Agent Full Name
0	Mandy	International Mandy	Mandy

Chat start reason	Chat end reason	Campaign	Engagement name
Skill reassign	Visitor disconnected	International	Default ARU chat box

Country	ISP	Organization	Device
India	Jio	Jio	MOBILE

Browser	Operating system	Chat start page	Chart start URL
Chrome 96.0.4664.45	Android	University courses at ARU | Anglia Ruskin University - ARU	https://aru.ac.uk/

Figure 3 – client metadata modes

3.4 Evaluation metrics

In this project, the performance of statistical models was gauged using sensitivity, or true positive rate. The reason this metric was chosen is different for each of the models used. In case of the neural network trained on the preliminary list of obstinate clients, its goal was to define strict criteria which would minimise the false positive rate; if the false positive rate was high, then many of the clients classified as obstinate in this project would not be classified as such by the customer support staff who first reported the phenomenon, and the project would thus investigate a different phenomenon to the one that was proposed. Therefore, minimising the false positive rate was essential. On the other hand, if the criteria were too strict, then the sample

size of clients classified as obstinate would be too small to conduct conclusive research. Hence, sensitivity constituted a balancing measure between metrics such as balanced accuracy, which allow too many false positives, and the inverse of false positive rate, which allows too few true positives.

With the *Objective 2 and 3* models, one of the underlying goals was to assist customer support staff with pre-emptive obstinate client detection in order to take prophylactic measures. However, while the nature of these prophylactic measures is outside the scope of this project, they would likely cost time and effort. Therefore, there is likely to be some cost to false positive predictions. On the other hand, there is no cost, as compared to the current situation, to false negative predictions ─ business resumes as normal. With early versions of tools such as these two models, cost minimisation usually takes priority over benefit maximisation. Hence, false positive minimisation was a priority. Like with the *Objective 1* model, though, the true positive rate cannot be too low, as then the tool will have too little practical use. Similarly, sensitivity proved to be the optimal balancing measure.

In *Objective 4*, it was assumed that the total amount of staff time taken up by a client is normally distributed. Therefore, a hypothesis test at the standard significance level of 5% given this assumption was chosen as the evaluation metric. It is important to note, however, this assumption is unproven, and requires further investigation.

Finally, in *Objective 5*, no evaluation metric was used because the volume of data was too small for any of the data to be allocated to testing without training. Therefore, no unbiased evaluation metric was possible. However, as more data is gathered, rigorous testing can be performed.

Chapter 4: Results

Objective 1:

A neural network was trained on the preliminary list of obstinate students, and was able to classify new clients with a sensitivity of 43% – 46% to 2 s.f. Given that the ratio of preliminarily obstinate to preliminarily non-obstinate clients was only 20% to 2 s.f., this suggests that there are identifiable texting patterns in voluminous clients, most of whom, according to reports, are obstinate. However, this conclusion is not definitive, as it is possible that confounding variables are responsible for most of the model's learning. Further investigation is needed for a definitive conclusion.

Objective 2:

A neural network that was trained on staff text data was able to detect obstinate clients with a similar sensitivity of 45% to 2 s.f., although the class distribution ratio, in this case, was only 1:25, yielding a true positive rate around 11 times higher than that of predictions by random chance. Nevertheless, in addition to the problem of confounding variables presented above, even if true reliable

detection is possible from staff input alone, the direction of causality isn't clear, and constitutes a potential area of future research.

Objective 3:

The best model performance was seen in a neural network trained on discrete metadata and via the means of a hyperparameter grid search: the model achieved a sensitivity score of 57% to 2 s.f. with the same class distribution ratio of 1:25. This result was not unexpected; the other models were not optimised via a grid search, and uncovering correlations between discrete variables is an easier task than uncovering patterns in human text. Nevertheless, this result shows that basic statistical tools to help customer support staff handle obstinate clients can begin to be deployed.

One area of future research may be principal component analysis using the same data to identify concrete factors that help predict obstinate behaviour.

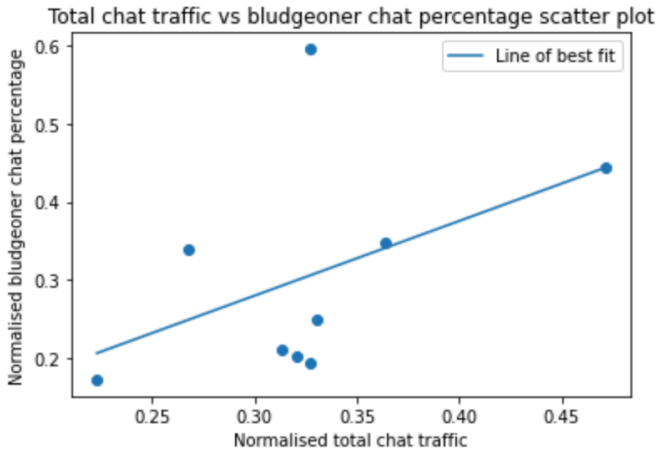
Objective 4:

Despite making up only 4% of all clients, obstinate clients contribute 33% of the total text volume, initiate 16% of the LiveChats, and take up 16% of staff time. Assuming a normal distribution of the amount of staff time taken up by students, the latter statistic proved to be a statistically significant result. However, testing whether modelling this random variable as being normally distributed is statistically justified or not is outside the scope of this project; hence, no definitive conclusions can be drawn at this time ─ for a definitive conclusion, the normality hypothesis needs to be tested.

Objective 5:

Two key results were found while completing *Objective 5*. First of all, it was found that the total number of LiveChats per month is moderately (0.40–0.55 depending on the runtime) correlated with the percentage of LiveChats initiated

by obstinate clients.




```
Correlation: 0.4596822579987044; 21% of the variance in bludgeoner chat percentage
explained by variance in total chat traffic.
```

Figure 4 – Total monthly LiveChats vs monthly obstinate client LiveChats scatter plot

One possible explanation is that there is a third variable, such as an increase in the number of assignment or admission deadlines, which both attracts more clients and encourages clients to make greater use of customer support. Another possible explanation is that obstinate clients tend to engage with customer support in groups, thus increasing the total number of clients. While there is presently no evidence for either of these hypotheses, this is a phenomenon that presents an interesting area of future research.

The second significant observation is that both total client traffic and obstinate client traffic display strong seasonality, as demonstrated by Figures 4 and 5. If this pattern of seasonality can be confirmed via follow-up testing, it would mean that obstinate client influxes are highly predictable and can be prepared for in advance. However, it is both conceptually plausible and is demonstrated by the figures that the period of seasonality is one year; if that is the case, then there has only been one period's worth of data, which is insufficient to draw any definitive conclusions.

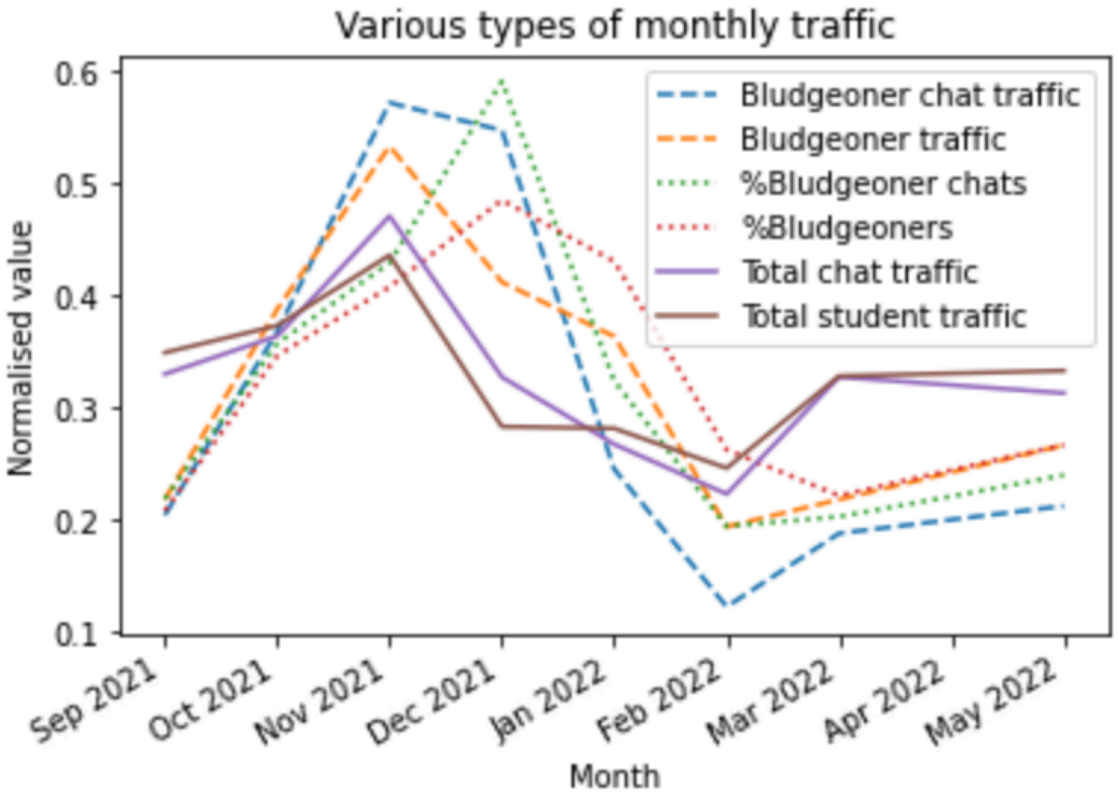


Figure 5 – Various measures of monthly traffic

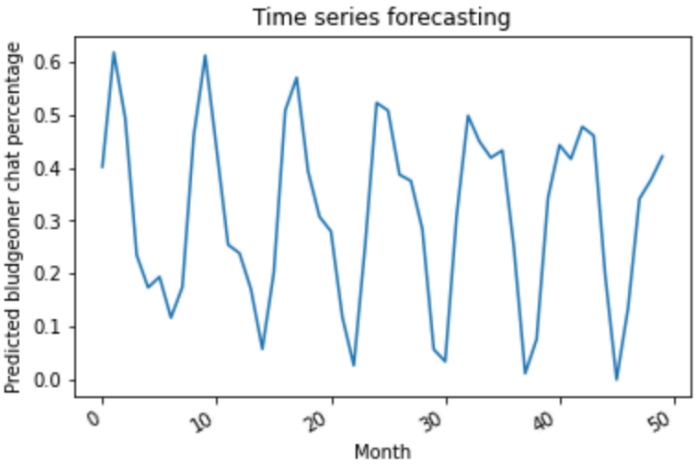


Figure 6 – Time series forecast

Chapter 5: Discussion

While the project has managed to produce some meaningful findings, as detailed in *Chapter 4: Results*, the scope of the project was significantly restricted by the resources available to carry it out. In particular:

1) LiveChat was only integrated into Anglia Ruskin University's website less than a year before the commencement of the project; therefore, the data that had been collected up until that point, and hence the data that was available to be used in the project, was highly limited. Consequently, a priori, any project findings were bound to have limited reliability.
2) The hardware used to train all the models described in *3.2 Research Framework* was confined to that of a single local Google Colab server, which imposed significant limitations on the machine learning methods that were available to be employed in the project. Therefore, a priori, it was not possible to aim to develop highly accurate and reliable statistical models.
3) The field of research of the project is almost entirely new; as such, there is a large range of candidate objects of investigation, all of which could not plausibly have been covered in a single project.
4) An additional consequence of factor 3) is that the volume of relevant literature is highly limited; therefore, few previous results could have been used to reduce the scope of the project, complete a comprehensive literature review, and compare the project's findings to.
5) Security protocol precluded potentially valuable data from being used in this project, even anonymously.

A consequence of factor 1) is that the final time series models are unlikely to be reliable as, given the conceptually plausible assumption that the period of seasonality is 12 months, less than one period's worth of data was used to train the time series, which is insufficient for reliable results. The impact of factor 2) has been detailed in the previous chapters, and includes only using a single classifier (multi-layer perceptron) to train all the classification models, the absence of hyperparameter grid search in 2 of the classification models, a reduced number of

grid search iterations, among others. A significant example of the impact of factor 3) is the fact that only 1 iteration of cross-validated training was performed for 2 of the models, with only 3 iterations being performed on one of the models; this precluded the possibility of reliable error bars. Additional such examples, as well as examples of the impact of factor 5), will be provided in *5.1 Next steps*.

In spite of the limitations, out of the 4 research hypotheses outlined in *1.2 Problem Background*, one was given a definitive conclusion, and for two more, the methodology for definitive testing has been presented. In particular:

1. It has conclusively (via cross-validation) been shown that obstinate clients can be detected from client metadata with moderate reliability (~57% sensitivity with a 1:25 class distribution ratio).
2. The direction of causality between staff text and obstinate behaviour by the client has not been established. The project's findings have not revealed a clear methodology for establishing said direction of causality. However, it has been shown that the latter can be predicted from the former with moderate reliability (~45% sensitivity with a 1:25 class distribution ratio).
3. Obstinate clients have been shown to take up a statistically significant proportion of the customer support staff time under the condition that the amount of staff time taken up per client is normally distributed. Hence, while a conclusive verdict has not been established regarding the original hypothesis, the testing process has been reduced to a test of whether or not the amount of staff time taken up per client is normally distributed.
4. A temporal pattern in the monthly prevalence of obstinate clients has been discovered, although not statistically confirmed. However, a clear methodology has been presented to test the hypothesis that a clear pattern exists: the same methodological needs to be applied but with more data collected.

5.1 Next steps

- As a consequence of factor 3), the following have not been investigated in this project, and constitute areas of potential future research:
    - Principal component analysis of the relationship between metadata and obstinate behaviour
    - Hypothesis testing to determine if the amount of staff time taken up per client is normally distributed
    - Hypothesis testing to determine if any of the hypotheses explaining the correlation between total client traffic and obstinate client prevalence outlined in *Chapter 4: Results*
    - A procedure to establish the direction of causality between staff text and obstinate behaviour
    - Building a lexicon, as defined in *2.1.2 Lexicon-based*, of individual terms and how closely associated they are with obstinate behaviour
- As a consequence of factor 5), the information on ultimate admission outcome was not available. Should that information be available, it would be useful to examine the relationship between ultimate admission outcome and obstinate behaviour as it would a) provide a measure of success of the staff's interaction strategy, and b) help staff set the optimal standards of communication. Hence an investigation into this relationship is a possible area of future research.
- As a consequence of factor 2), the accuracy and reliability of the models built in this project could be improved. Building more accurate and reliable versions of the models built in this project constitutes another possible area of future research.

Chapter 6: Conclusion

The phenomenon of obstinate clients is almost entirely undocumented in existing literature and, as a result, leaves exciting areas of research open to their undertaking. This paper aimed to address the most fundamental of these areas, with both the short-term goal of providing immediate assistance to customer support staff at ARU and around the world and the long-term goal of facilitating further research on the topic.

The findings of this paper have shown that there are legitimate patterns surrounding the topic which, if discovered, can provide great insight into not only obstinate clients and how to handle them, but potentially into other areas of electronic communication analysis. Moreover, early versions of statistical tools have been developed to assist customer support staff in dealing with obstinate clients, and have been demonstrated to yield satisfactory, although limited performance. As such, they are ready to be experimentally deployed in the workplace and are expected to both reduce the amount of stress experienced by staff and the quality of support received by the average client.

While some of the most intriguing hypotheses proposed in this project haven't been conclusively corroborated or refuted, the hope is that, as more data is gathered and more research into the topic is carried out, the problem of their veracity, as well as many other interesting yet unformulated problems surrounding the topic, will be eventually resolved. Similarly, while the statistical tools developed in this project have already been shown to perform at a satisfactory level, they can be re-used with additional data and/or for follow-up projects for deeper insight and increased reliability.

Reference list

[1] R. Mano, G. Mesch, 2012, "E-mail and work performance", doi:10.4018/978-1-4666-0315-8.ch009.

[2] D. Rathee, S. Mann, 2022, "Detection of E-Mail Phishing Attacks – using Machine Learning and Deep Learning", in International Journal of Computer Applications 183(47):1-7, doi:10.5120/ijca2022921868.

[3] I. Fette, N. Sade, A. Tomasic, 2007, "Learning to Detect Phishing Emails", in WWW '07: Proceedings of the 16th international conference on World Wide Web, pp. 649-656, doi:10.1145/1242572.1242660.

[4] Y. Fang, C. Zhang, C. Huang, L. Liu and Y. Yang, 2019, "Phishing Email Detection Using Improved RCNN Model With Multilevel Vectors and Attention Mechanism," in IEEE Access, vol. 7, pp. 56329-56340, doi: 10.1109/ACCESS.2019.2913705.

[5] Lamba, Manika & Margam, Madhusudhan, 2022, “Sentiment Analysis”, pp. 10, doi: 1007/978-3-030-85085-2_7.

[6] R. H. Shumway, D. S. Stoffer, 2017, "Time Series Analysis and Its Applications: With R Examples", Springer Texts in Statistics 4th Ed. (eBook), pp. 75-154, doi: 10.1007/978-3-319-52452-8.

[7] Hoaglin, David, 2003, “John W. Tukey and Data Analysis”, Statistical Science, pp. 18, doi: 10.1214/ss/1076102418.

[8] Willett, Peter, 2006, “The Porter stemming algorithm: Then and now”, Program electronic library and information systems, pp. 40, doi: 10.1108/00330330610681295.

[9] Qader, Wisam & M. Ameen, Musa & Ahmed, Bilal, 2019, “An Overview of Bag of Words;Importance, Implementation, Applications, and Challenges”, pp. 200-204, doi: 10.1109/IEC47844.2019.8950616.

[10] David G.T. Barrett, Felix Hill, Adam Santoro, Ari S. Morcos, Timothy Lillicrap, 2018, “Measuring abstract reasoning in neural networks”, arXiv:1807.04225 [cs.LG], pp. 8, doi: 10.48550/arXiv.1807.04225.

[11] Potdar, Kedar & Pardawala, Taher & Pai, Chinmay, 2017, “A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers”, International Journal of Computer Applications, 175, pp 7-9, doi: 10.5120/ijca2017915495.

[12] E. Kreyszig, 1979, “Advanced Engineering Mathematics (Fourth ed.)”, Wiley, pp. 880, eq. 5, ISBN 0-471-02140-7.

[13] Lamba, Manika & Margam, Madhusudhan, 2022, “Sentiment Analysis”, doi: 10.1007/978-3-030-85085-2_7

[14] Rodrigues, Ramon & Camilo-Junior et al, 2018, “A Taxonomy for Sentiment Analysis Field”, International Journal of Web Information Systems, pp. 197, doi: 10.1108/IJWIS-07-2017-0048.

[15] Joshi and Itkat, 2014, “A survey on feature level sentiment analysis”, International Journal of Computer Science and Information Technologies, Vol. 5 No. 1, pp. 5422-5425.

[16] N. Q. Do et al, 2022, "Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions," in IEEE Access, vol. 10, pp. 36429-36463, doi: 10.1109/ACCESS.2022.3151903.

[17] Alsaed, Zaina & Eleyan, Derar, 2021, “APPROACHES TO CYBERBULLYING DETECTION ON SOCIAL NETWORKS: A SURVEY”, Journal of Theoretical and Applied Information Technology, 99, pp. 3096-3109