Insight from Data Analytics in a Facilities Management Company

Daniel Walker[1], Martin Ruane[2], Jaume Bacardit[3], Shirley Coleman[4]
[1]*School of Mathematics, Statistics and Physics, Newcastle University, Newcastle upon Tyne, UK & Engie UK.*
[2]*Engie UK*
[3]*School of Computing, Newcastle University, Newcastle upon Tyne, UK.*
[4]*School of Mathematics, Statistics and Physics, Newcastle University, Newcastle upon Tyne, UK.*

Abstract

Facilities management of large public services is commonly outsourced to a specialist provider. Extensive data is collected on a wide range of operational tasks. Often, limited use is made of this data beyond operational considerations. Increasing availability of sophisticated data integration systems makes it possible to develop bespoke data science solutions to improve the efficiency, speed and accuracy of commonly encountered tasks.

A Knowledge Transfer Partnership (a funding scheme of Innovate UK) was set-up between Newcastle University and a large facilities management provider. Within this project, data science has been effectively applied to improve service delivery in three different case studies: (1) designing routing algorithms for scheduling of staff allocation to jobs, (2) using natural language processing for automatically raising work orders from help desk emails, (3) development of algorithms to optimise the placement of charging points for the transition of fleets of commercial vehicles from diesel to electric powered. For each case study, we show how the data science work has contributed to improvement of facilities management by: (1) maximising the productivity of engineers, (2) reducing the time taken to process help desk emails, (3) creating a data driven method to provide objective decision support for charger placement, respectively.

The research in this paper is based on a contract with a local authority where the facilities management company provides planned and reactive maintenance for a range of assets in corporate buildings, hospitals and schools. In addition to describing the data science solutions, we will also discuss issues around knowledge transfer from data scientist to operational staff and how change management processes were employed in this project to embed the new working practices.

1 Introduction

Often administrations choose to outsource the facilities management (FM) of public services to a specialist provider. FM includes ongoing maintenance of buildings and sites. Planned maintenance needs to be undertaken at fixed intervals and reactive work requires dynamic response to faults of varying urgency. Computer-aided facilities management (CAFM) systems are used to log jobs and monitor performance against key performance indicators (KPIs). The data in these systems is an untapped resource that can be used to solve some of the sector's most challenging problems such as ineffective job scheduling and transitioning to zero carbon.

Embarking on data science requires initial effort in acquiring expertise and developing it in terms of the unique business needs before there are any financial gains. Although techniques that would substantially benefit companies are available, small to medium enterprises (SMEs) and companies in the FM sector are reluctant to invest the time and resources to make use of them as discussed in.[1] The UK government helps implement new skills and capabilities by providing 50 per cent funding via Innovate UK to set up a Knowledge Transfer Partnership (KTP) to support a 1 to 3 year project in which a post graduate research associate is employed by a partner university but works full-time in the company on a substantial project likely to have demonstrable impacts and improve the profitability of the company. The KTP project needs to provide the academics with new research experience and useful ideas and data for teaching as well as embedding a new capability in the company and improving the productivity of the UK.

A KTP between Newcastle University and a multinational company has provided the opportunity to explore the applications of data science in the conduct of a contract between the company and a local authority. The company provides planned and reactive maintenance for a range of assets in corporate buildings, hospitals and schools. The KTP mechanism significantly reduces company business risk of embracing new methods while exploiting knowledge and opportunities from subject-matter experts and their broad access to multiple industry models and research ideas. This approach to managing new methodologies serves the company governance decision-making model very effectively and is novel in the FM sector.

Data routinely generated in this sector includes details of maintenance tasks, type of job and required skills, time taken on the job, travel time, location, site, spatial and temporal data. Data arises from different sources, some being logged automatically such as geographical location, some being provided by operatives. Data is owned by different departments, stored in different formats and using different encodings. These are common challenges in any big data project and merit due consideration. This KTP aims to apply data science methods to produce outcomes of value to key decision-makers. Clearly it is pivotal that company personnel are aware of the benefits and the

KTP has produced quantified outputs in a number of different areas, has contributed to a common data hub and provided regular group briefing and training sessions.

Increasing constraints on expenditure and the need to show value for money mean that there is considerable interest in how facilities management is undertaken. In addition, the hazards of diesel vehicle emissions have led the company to set targets for environmental improvements including the change to using electric vehicles as an alternative to vehicles that rely solely on fossil fuels. Datasets compiled from the Internet of Things and Industry 4.0 result in massive amounts of data being available. A data science approach applying statistical thinking and utilising these extensive datasets has enabled improvements to be made in typical operational and maintenance issues encountered throughout the service sector.

FM teams are subject to detailed scrutiny, and expectations in terms of performance and efficiency are increasing all the time. This has promoted a discriminating focus on data and an openness to developing new ways of creating value from the data. Suitably analysed business data can provide FM teams with evidence to support strategic changes, recommend where value can be added and sea-changes can be introduced to the working pattern, thereby paving the way for dramatic process improvement.

Apart from work on smart buildings,[2] data science has barely been explored within FM and it can benefit from experiences and work produced outside the sector. Data science is not a core competency for service sector managers and ways of facilitating knowledge transfer from data scientist to operational staff have been evolved as part of the project. As new working practices evolve, change management is an important point.

As this paper will show, an effective application of data science to FM is a highly interdisciplinary endeavour that uses techniques from a broad range of disciplines including statistical analysis, machine learning, process control and optimisation, and applied to descriptive, predictive and prescriptive analytics

1.1 Roadmap

The rest of the manuscript consists of three use cases in which data science has utilised FM data to provide benefits to the various stakeholders in the sector. Each example consists of: (1) the background of the use case with details of the business driver and the data; (2) the approach and techniques used including details of the hardware and software used for the data science solution; and (3) the results of the analysis and business value derived. Section 2 describes a bespoke approach to job scheduling, section 3 applies natural language processing to help desk emails and section 4 considers the placing of electric vehicle charging points. Section 5 discusses the challenges involved in applying data analytics and implementing new methods ensuring that solutions are continued and sustainable within the business partners, this includes creating coding clubs and a new data analytics focused unit within the company. We then conclude with a summary of lessons learned and management implications.

2 Use Case 1: Job Scheduling

2.1 Background & Data

This use case focuses on a proof of concept with one of the FM company's client contracts. The main aim is to create a scheduling tool which can use data from the Computer-Aided Facilities Management (CAFM) system to automatically schedule planned maintenance tasks to engineers. Each month, a report is created from the CAFM system that contains a list of tasks to be completed the following month. The owners of the tasks are supervisors of trade teams who have the necessary skills to complete the maintenance tasks. For example, the gas team supervisor might have some annual boiler service and gas shut off valve inspections. Typically, these tasks have to be completed within the month, although some tasks repeat weekly within the month.

In the old process, the supervisor for each trade team would randomly assign their tasks to each member of the team. For each task, the supervisor would raise a work order which would be sent to the engineer's handheld device. No date or time would be allocated to the task and so it was the responsibility of the engineer to plan when the service would be completed. The engineer starts a timer on the device when the work is started, and stops it when the work is finished. These process times are recorded in the CAFM system.

Some of the motivations for creating a scheduling system are listed and discussed below:

- Creating transparency of data for performance management
- Reducing travelling by scheduling efficient routes
- Being more predictable and provide a better service to the client
- Creating admin efficiencies through automation

Prior to using a data-driven method for scheduling work, there was very little performance management of the engineers and the work they were doing. One desirable side effect of implementing a scheduling system is that the data is monitored on a dashboard to understand process times which gives supervisors the ability to track progress against tasks and investigate anomalies before they result in failures of a key performance indicator (KPI). One of the company's top priorities is reducing environmental damage and transitioning to zero carbon. Hence, scheduling efficient routes is a key motivator for introducing a scheduling system to prevent unnecessary travel caused by lack of planning from the engineer. Having a planned schedule allows the company to make appointments with the client. This is better service for the client and also prevents situations where the engineers 'cold call' to complete some work and are refused access, which is a waste of valuable engineer time. Lastly, automating the scheduling of the work removes headaches for supervisors by removing mundane tasks from their workload and freeing them to focus on managing team performance.

To produce feasible schedules, it is important to understand some of the input variables that affect the process. The key variables considered were the attributes associated with each job such as the type, location and due date; the historic data on process times for each job type, with consideration given to the mean and variance; any constraints affecting the possible days or times that a task can be done; distances and travelling time between sites. The following data sources were used to gather this information.

- **Inspection Report:** This report is a list of jobs that are due to be completed within a particular date range. Trades operate independently and so one report is produced per trade. The data comes from the CAFM system and is typically exported as a csv file. Usually, one month of data is extracted and the number of jobs is typically in the range of 30-300. This is dependent on the time of year and trade. The data contains information about each inspection such as the site, type and due date.
- **Historic Labour Data:** The expected process time for each job is required in order to schedule the work. Engineers typically use a personal device with an application for recording labour hours against jobs. This data is stored in the CAFM system and can be extracted as a labour report. The latest 12 months of jobs were extracted in order to observe timings. The timings are dependent on the job type, the site and, to a lesser extent, the engineer. The labour records were grouped by site and type, then the median was taken to be the estimated process time. Most of the tasks are monthly and so the median is of 12 observations. Some tasks are less frequent and so there are fewer data points. The impact of the different engineers has been ignored as there was not enough data to understand the variation (not every engineer has done every task at every site). In total, 19,969 tasks were analysed to understand process times.
- **Subject Matter Experts:** Supervisors were consulted to help understand scheduling constraints. These constraints are business rules that determine times and days that certain jobs can be scheduled. For example, some buildings can only provide access for the engineers to complete the work on days when the caretaker is available. There are many occasions where access to a building is not granted due to the engineer showing up unannounced at an inconvenient time. Hence, it is important to account for all the building access constraints when scheduling the work. This has proved particularly challenging with the global pandemic as these constraints and demands are ever changing with building users cautious to grant access at any time.
- **Travel Time Matrix APIs:** In order to schedule effectively, the time taken to travel between each site must be known. There are a number of different API services that can

provide this information. For this case study, it was deemed that Open Route Service API was the most suitable as it allowed for up to 500 free requests per day. The API uses open street map data.[3]

2.2 Methodology

In this section, two approaches to modelling the problem are briefly discussed and the chosen model is defined mathematically. The company has a task scheduling problem to solve. It has to allocate resource between a variety of tasks and create a sequence of visits so that engineer travelling is minimised and time and day constraints are adhered to. The problem is considered to be an optimisation problem best represented by either: the vehicle routing problem with time windows (VRPTW) or the multiple-agent, maximum collection problem with time-dependent rewards (MAMCPTD). The two approaches are discussed briefly below.

2.2.1 MAMCPTD

The MAMCPTD[4] is a variant of the multiple-agent maximum collection problem (MAMCP) defined by Butt & Cavalier.[5] In MAMCPTD, there are multiple agents deployed to collect linearly decreasing rewards over time. The objective is to maximise total surplus (total reward collected minus total travel cost) by routing multiple agents from a central depot. The rewards are internal values used by the algorithm to determine which days are most satisfactory for the work to be done rather than monetary rewards. The algorithm may be criticised for its subjectivity in assigning rewards to tasks. The company's problem is quite similar to that of the MAMCPTD. Some differences to overcome are highlighted below:

- In MAMCPTD, the time taken to complete a task cannot be greater than the working hours per day. In the case of the company, this situation occurs occasionally.
- In MAMCPTD, a single tour is determined for each working day. In the case of the company, there are multiple tours per day since there are multiple engineers working every day.
- In MAMCPTD, there are no considerations of time windows when a task can be completed within a day. This is a key difference as the company has many time constraints it needs to meet. For example, work done in school kitchens can only be done after the school lunch hour.

2.2.2 VRPTW

The VRPTW[6] is a variant of the vehicle routing problem, the multiple agent version of the travelling salesman problem. The aim is to route agents from a central depot to a set of customers who need attending to in given time windows. The routes should minimise the total amount of travelling over all agents. Again, this problem fits the problem of the company well. The differences are noted below:

- In VRPTW, there is no formulation of multiple days and in the case of the company, it wants to schedule a month of work at once.
- A site can only be visited once in the VRPTW. This might not be the case for the company who might require multiple visits to a site.

It was concluded that the simplest model to adapt to the company's problem would be the VRPTW so this is the chosen model.

2.2.3 Mathematical Formulation of VRPTW

Below is a general mathematical formulation[7] of the VRPTW and a table of notation.

Notation	Description
C	Set of customers
V	Fleet of homogeneous vehicles
$G(V, C)$	Directed graph
N	Set of vertices $\{0, ..., n\}$ 0 is the depot and $\{1, .., n\}$ are customers
A	Set of arcs (i, j)
c_{ij}	Time in minutes to travel from i to j
p_i	Service time for i
t_{ij}	Service time at i plus time to travel to j $(p_i + c_{ij})$
$[a_i, b_i]$	Time window for the engineer to begin work at i
x_{ijk}	Binary decision variable = 1 if vehicle k traverses (i, j)
s_{ik}	Relative time that vehicle k begins service of i

Table 1: Table of notation for reference

The VRPTW is defined as a set of customers, C, a fleet of homogeneous vehicles, V and a directed graph, $G(V, C)$. In our case, we will assume the customers to be the individual servicing tasks. The graph has $|C| + 1$ vertices with the tasks denoted $1, 2, ..., n$ and an extra vertex for the depot, denoted by 0. Hence the total set of vertices $N = 0, 1, ..., n$.

The objective is to minimise the travel time. The set of arcs, A, represents the connections between vertices. For each arc, (i, j) we associate a cost, c_{ij}, represented by the journey distance, and a time, t_{ij} which includes the journey from i to j and a service time for task i.

Each task, has a time window in which the job must be started. This may be due to building restrictions or contractual compliance. Time windows are denoted as $[a_i, b_i]$ so that the vehicle must arrive before b_i and cannot begin the service before a_i, although they may arrive before and wait.

We have two decision variables x_{ijk} and s_{ik}. Variable x_{ijk} is a binary variable which is equal to 1 if arc (i, j) is traversed by vehicle k in the optimal solution and equal to 0 otherwise for $i \neq j, i \neq n + 1, j \neq 0$.

Variable s_{ik} represents the time that vehicle k begins the service of asset i; if vehicle k does not service asset i, then s_{ik} does not mean anything. We take $a_0 = 0$ and therefore, $s_{0k} = 0, \forall k$.

Using the above notation, we can define the problem as follows:

$$min \sum_{k \in V} \sum_{i \in N} \sum_{\substack{j \in N \setminus \{0\} \\ j \neq i}} c_{ij} x_{ijk} \tag{1}$$

subject to the following constraints:

$$\sum_{k \in V} \sum_{j \in N} x_{ijk} = 1 \ \ \forall i \in C \tag{2}$$

$$\sum_{j \in N} x_{0jk} = 1 \ \ \forall k \in V \tag{3}$$

$$\sum_{i \in N} x_{ihk} \ - \ \sum_{j \in N} x_{hjk} = 0 \ \ \forall h \in C, \ \ \forall k \in V \tag{4}$$

$$\sum_{i \in N} x_{i,n+1,k} = 1 \ \ \forall k \in V \tag{5}$$

$$s_{ik} \ + \ t_{ij} \ - \ K(1 - x_{ijk}) \ \leq \ s_{jk} \ \ \forall i, j \in N, \ \forall k \in V \tag{6}$$

$$a_i \ \leq \ s_{ik} \ \leq \ b_i, \ \ \forall i \in N, \ \forall k \in V \tag{7}$$

$$x_{ijk} \ \in \ 0, 1 \ \ \forall i, j \in N, \ \forall k \in V \tag{8}$$

Constraint (2) ensures that each asset is serviced exactly once. Constraints (3),(4) and (5) state that every vehicle leaves the depot, 0, leaving each asset once serviced and finally concludes the journey at the depot 0. For a vehicle k travelling from asset i to asset j, (6) states that k cannot arrive at j before $s_{ik} + t_{ij}$ which is the time that it services i plus the time taken to service i and travel to j. Here, K is a large scalar. Constraint (7) forces the time windows to be observed and (8) are the integrality constraints. An unused vehicle is modelled by the route (0,0).

2.2.4 Implementation & Adaptations

This type of problem is well researched[7] and, typically there are two differing approaches to solving it. Exact solution methods find the optimal solution of the objective function whereas heuristic approaches use an initial solution strategy and metaheuristics to find an an approximate solution to the problem. The size of the problem heavily impacts which approach to take as exact approaches can be computationally very expensive as the number of nodes increase. The complexity of the problem is discussed here.[8] In this use case, it is required that a problem size of at least 100 nodes can be solved in a short amount of time. This is because users may want to re-run schedules frequently and visualise routes quickly. Some trial and error with exact methods was conducted. The problem was coded as a mixed integer programme using the CBC open source solver[9] and tested on problems of increasing size. The experiments were tested on a Lenovo ThinkPad T460s with an intel i5 CPU and 8GB of RAM. The wall-clock time was recorded in seconds. The scalability of the solver can be inferred from the data presented in figure 1.

Figure 1 about here.

Based on the data in figure 1, it was apparent that the CBC[9] solver would be incapable of solving a problem with more than a hundred sites in a reasonable time, especially considering that the schedules need to be run daily. Some commercial solvers may have been able to provide a much faster solution but there was no budget available to test this hypothesis. Hence, the search began for an open source heuristic solver.

Google Operation Research (OR) Tools offer an extensive package for solving optimisation problems in various languages including Python.[10] Specifically, they offer a library for solving routing problems, mainly different variants on the travelling salesman problem and its multi-agent equivalent, the vehicle routing problem.

Google OR-tools[10] provides the code for solving a dummy example of the VRPTW. It was able to solve a problem with 100 sites in a few seconds making it a suitable candidate solver. The following adjustments to the code were made to generalise the model for this use case. Allow for:

- A non-zero servicing time at each job
- Scheduling multiple days
- Multiple depots or home start locations
- Different shift patterns and resource availability
- Multiple agents attending one job

Google OR-tools[10] offer various first solution strategies and metaheuristics to solve the problem. An initial solution is found using a greedy algorithm and a local search method aims to escape local minima. Based on a subjective view of some trial and error with different combinations of strategies, there was very little difference in the outputs. The chosen strategies were a parallel cheapest insertion with guided local search metaheuristic. Some further work could be done to objectively test which strategies performed the best for this use case, potentially with some designed experiments.

2.3 Results

An intermediate process is in place currently while the tool is being trialled. This process and the proposed future process can be seen in figure 2.

Figure 2 about here.

The data has been tracked through this trial period using some aspects of statistical process control (SPC). We are interested in the variation of process times for completing planned maintenance tasks as this directly affects the schedules. If process times are stable and predictable then compliance with the schedule will be higher. Where there are diversions from the schedule, it is important to understand why these diversions happened. Shewhart charts (I-charts) were use for monitoring gas boiler servicing times. The application of these control charts has two phases. In phase one, data is collected and analysed retrospectively. Trial control limits are constructed to determine if the process has been in control during the time that the data was collected and to see whether reliable control limits can be established for phase 2 monitoring.[11] In this phase, data points outside the control limits are removed and new trial control limits are constructed until we have a set of in control data. Figure 3 is an example of the raw data collected on gas boiler servicing times in phase 1. The graph shows a subset of the data gathered. In total, there were 43 data points.

Figure 3 about here.

Observing control charts for process times allows for easier identification of special cause variation. In figure 3, there are two cases where the process time exceeded the upper control limit. By addressing these cases, issues have been uncovered with engineer practices and audit trails:

- The first data point highlighted an issue where multiple gas boilers were on site but only one of those existed in the asset database. Hence only one job was raised and the engineer recorded multiple inspections on one labour record. Following this observation, the other assets were included in the asset register.
- The second data point was a high process time because the engineer noticed a fault during the standard maintenance task and proceeded to fix the boiler while recording his labour on the inspection code. The task of fixing a fault is deemed to be reactive work that the company is obliged to charge the client for but recording time against an inspection code meant that they were unable to charge the client. Hence, uncovering this behaviour has a potential positive cost impact to the company if the engineers can be trained to avoid doing this in future.

These two points were removed from the analysis in phase 1 and the control limits were recalculated. The remaining data points were all within the new control limits. In phase 2, we now have information on the average run length and the gas boiler servicing times will be monitored.[11] Currently, there is not enough data to provide results on phase 2 of this analysis.

These graphs have been shared with the contract through the use of a PowerBI dashboard. The benefit of using this tool is that it is part of the Office 365 group and so it can be easily and securely shared with anyone who would benefit from seeing it. The use of dashboards for reporting data is nothing new but, in this use case, it played an important role in providing transparency of data to all stakeholders where previously there was none. The transparency is important because it gives management a means for driving process improvement and allows them to do it quickly by observing live data. Previously, any data analysis was conducted in end of month reports, by which time it is too late to address any issues that might affect the key performance indicators.

The impact of Covid-19 meant that the trial was interrupted on multiple occasions as the team worked to deal with the changing climate they were facing. New arrangements had to be made in schools and many of the corporate buildings were either shut or refusing access to complete servicing. Baseline data was gathered prior to using the tool, and there is also some data to compare to in months where the trial has been uninterrupted. The results and benefits are summarised in Table 2.

Table 2 about here.

Many of the results are qualitative. The conclusion section provides lessons learned and what could have been done to provide more quantitative results.

2.4 Conclusions

This use case provides many potential benefits that are desirable to management and so it has been relatively easy to sell it to the business. For example: the concept of efficient routing to reduce travel and carbon footprint is aligned well with company strategy; monitoring process times and aiming to reduce variation gives management some clarity on the capacity of the team and how well utilised the engineers are; and scheduling work to engineers rather than having them manage their own workload provides transparency on the activities that are being done in the company and its clients, providing a basis for better performance management. In practice, realising these benefits and tracking them has proven difficult. There are often cases where real world factors are in contradiction to the vehicle routing problem that is being solved. An example of this is the access to buildings during the covid-19 pandemic. Building users were working from home which meant that engineers had to frequently re-route to various locations to obtain keys. Another example might be emergency jobs where engineers are required to attend immediately to a reactive job that was not pre-scheduled. In some cases there is resistance to comply from engineers or supervisors who have grown comfortable with working in a certain manner for many years. Currently, the number of jobs that are completed as according to schedule is as low as 50% most months and it is hard to pinpoint all the reasons for that lack of compliance. The same lack of compliance makes it difficult to track some of the benefits as there are few examples of days when the old method can be directly compared to the scheduling method. A priority for this project moving forward is to understand the lack of compliance and to collect more quantitative data for validation. Working more closely with the supervisors and engineers to gather feedback will help understand the compliance issues and more clearly defined performance metrics will help provide better quantitative results. A summary of lessons learned is below:

- Conduct more extensive research on potential models that fit the use case
- Clearly define the metrics that will be improved and create a plan to track and compare to a baseline
- Work closer with end-users to understand requirements and where the model does not match reality

3 Use Case 2: NLP with Help Desk Emails

3.1 Background & Data

The second use case involves natural language processing (NLP) of help desk emails. Planned maintenance as discussed in section 2 is one work stream that is contractual and predictable. The other main source of work comes from reactive jobs where the client reports faults with an asset and the company has to respond. Depending on the severity of the fault, the job may be classified as emergency, urgent or routine. Service level agreements are in place to ensure the faults are addressed in an appropriate time frame. These faults can be reported via a number of channels, the most frequent of which is an email to the company help desk. The operatives on the help desk need to constantly triage the inbox for the most important tasks and create tickets in the system for the planning team to pick up and assign to trade teams. The tickets should contain sufficient information to allow the planning team to assign an engineer to carry out the work. The aim of this use case is to monitor the help desk emails and use natural language processing techniques to automatically raise the help tickets in the CAFM system. Historic emails and their corresponding tickets are used as training data for an open source deep learning named entity recognition model. To create automatic tickets, there are primarily three entities to extract; site, asset and fault. This information tells us the type of equipment that needs fixing, the problem with the equipment, and the site where the issue is.

There are around nine hundred emails to the help desk per month requesting reactive maintenance. To capture a baseline of how much time is spent raising tickets from these emails, an exercise was undertaken to spend a full afternoon observing an operative on the help desk, recording the

times they spent raising tickets. The mean time taken to deal with each of these requests was a fraction under five minutes. Assuming an average process time of five minutes, and an average of 900 emails per month, this is 4500 minutes or 75 hours per month. In terms of full-time employee (FTE) time, this equates to 0.5 FTE on this contract, assuming 150 hour months. Automating a proportion of these emails could provide a small but significant saving to the contract. Further to this, we anticipate that the model will be useful across the business where we have similar contracts. Just in the north of England there are at least 5 similar sized contracts.

3.2 Methodology

With a wealth of information available in historic emails, some exploratory analyses were conducted to try and get a broad picture of the content. When an email requesting work reaches the help desk, the operative must assign it to a pre-defined class, depending on what the issue is. These classes are ever evolving and many of them are not used. Latent Dirichlet Allocation (LDA),[12] is an unsupervised topic modelling method which can identify themes, or topics, within a large set of text data. LDA was used to identify topics within the emails which were then compared with the existing classes.

The SKLEARN[13] package in Python was used to conduct LDA and visualise the topics. One caveat with LDA is that the user must specify a number of topics as a parameter for the model, which is a difficult task. Coherence score[14] is a measure of how interpretable the topics are. In order to optimise the coherence score of the model, a grid search method is used. Grid search methods are used for tuning model parameters relative to some metric, in this case coherence score. A parameter search space of between 2 and 15 topics was defined and LDA models created for each parameter possibility. For each model, the coherence score was calculated and the optimum parameter value was chosen based on the model with the highest coherence score.

Figure 4 about here.

Figure 4 is a visualisation of the topics of the help desk emails based on the optimum number obtained from the grid search. On the right is a list of the most salient terms in one of the topics. The visualisation is rendered in html and, in practice the user can dynamically choose to look at any of the topics, a screenshot of one topic is shown in figure 4. Through using this technique, common themes were observed with emails that come into the help desk which helped to inform a new classification system for the types of the jobs that arise.

Creating a help ticket from the unstructured email is essentially a named entity recognition task. The approach was to review the existing research in this area to find a model that could be suitably adapted to our problem. There are many open source named entity recognition systems that could be used as a framework to tackle this specific problem. Four of the most common are reviewed here.[15] For the purpose of this task, the spaCy NLP software package was chosen as it is the current industry standard for NLP and the software library with the smallest entry barrier in terms of deploying NLP in a commercial setting. The architecture of the model and details of the API as well as code examples are given in.[16]

spaCy uses at its core a multi-task convolutional neural network model that is trained on OntoNotes corpus which is a very large dataset of annotated text.[17] This model drives a pipeline of NLP tasks (e.g. tokenisation) and gives the user access to the output. See the model pipeline in Figure 5.

Figure 5 about here.

The named entity recognition model is capable of recognising the entities named in Figure 6.

Figure 6 about here.

Emails from the historic data were tested on spaCy's default named entity recognition model to see whether it was able to extract the names of the company's sites from the emails as a location or building entity. Actually, the default model performs very poorly for this situation as the emails contain lots of local slang and the sites and buildings that are serviced by the company are highly variable. For example, there are schools, caravan parks, libraries and corporate buildings. Fortunately, SpaCy has the option to create your own training data with custom entities and use this to train a blank version of the default model. This provided the difficult task of creating some training data. An example of the format of the training data is shown in table 3. The original email is annotated with the locations and type of entities to be extracted. An existing annotation tool from the open source community was used to speed up this process.[18]

Table 3 about here.

The annotations in table 3 show the position and type of the entity so that the model can learn which entity is being referred to and where it is in the text. In the example above, the name of the site is mentioned in character positions 29 to 38.

For this custom model, training data was annotated with 4 custom entities:

1. **Site:** The location or building where the fault has been reported
2. **Asset:** If the specific asset affected was mentioned in the email, it was tagged. e.g. boiler, water pipe, radiator.
3. **Fault:** Any words that described the nature of the fault were tagged. e.g. leaking, broken, missing.
4. **Trade:** If there was a mention of a specific trade type, this was tagged. e.g. plumber, electrician, joiner.

The time taken for this annotation process is approximately 30 seconds per email when using the annotation tool. The tool outputs a json file in the format required for training the spaCy deep learning model. This made the process of training the model quite straight-forward as the spaCy documentation shows some sample code for doing this. The model follows the pipeline: parsing and tokenising the text, modelling the dependencies between words. This means that the model is contextualising sentences and not just choosing entities based on the individual words. An example of dependency parsing is shown in Figure 7.

Figure 7 about here.

Despite having the annotation tool, preparing the data for training the model was quite laborious and hence it is important to have some metrics for understanding how good the model is as a function of the size of the training set. This way, only the minimum number of emails have to be annotated to reach an acceptable model. Typically deep learning models require large amounts of data although this paper[19] suggests that multi-task models are also suited to smaller data sets.

The training process took around twenty minutes to run on a standard industry laptop with an i5 processor and 8GB of RAM. spaCy allows you to save the model in a data directory so that it can be reloaded every time it needs to be used. Further training only needs to happen if more data has been annotated.

See figure 8 which shows the process for annotation, training, and eventual use of the named entity model.

Figure 8 about here.

3.3 Model Validation

The named entity recognition (NER) model was validated by using 5-fold cross validation. Cross-validation involves shuffling the labelled data and splitting it into 5 groups. The model is trained 5 times, each time holding back a different group of data as a validation set. Each of the trained models makes predictions on its validation set and the precision, recall and f1 scores are gathered. The mean precision, recall and f1 scores are used from the set of 5 scores. For more information on cross validation see this paper.[20]

The precision, recall and f1 scores for the named entity model are calculated as follows. The number of true positives, false positives and false negatives are counted for each entity type.

- **True Positive (TP):** The model predicts that a string is an entity, the labelled data agrees with the exact string.
- **False Positive (FP):** The model predicts that a string is an entity, the labelled data suggest that this string is not exactly that entity
- **False Negative (FN):** The model does not predict that a string is an entity, the labelled data suggests that this string is that entity.

$$Precision = \frac{TP}{(TP + FP)} \tag{9}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{10}$$

The f1 score is defined as the harmonic mean of precision and recall. This gives an overall picture of performance when there is a trade-off between precision and recall scores. Further, we would like to plot these scores against different training set sizes to see how the model performs as the size of the training data increases. Figure 9 shows the results of this task where each point on the graphs represents the mean precision,recall and f1 scores from a 5-fold cross-validation for a given training set size.

Figure 9 about here.

The most important entity to extract is the site entity. It is imperative that the correct site is extracted so that there is no situation where someone is sent to the wrong site. There is a clear positive trend between the training set size and the precision, recall and f1 scores for the site entity. These scores are around 80 but they do not appear to have plateaued so this score could almost certainly be improved by annotating more training data. The model performs well on the trade entity. This is probably because there is only a small set of trade names and not many variations in terminology. For the asset and fault entities, the opposite is true; there is a vast number of different assets and faults to report and various ways that they can be expressed.

3.4 Conclusions

This section addresses the outcomes and lessons learned from undertaking natural language processing of emails. The aim was to provide a proof of concept to help decide whether the methodology could usefully be applied. There are some promising results from the named entity recognition model but there are lots of improvements to be made and barriers to overcome before we can implement a solution. These can be summarised as follows:

- Increasing the size of the training data to improve model scores.
- Standardising and improving the approach to data annotation.
- Creating methods to classify problem type.
- Implementation: integration to email accounts etc.

- Filtering emails that are not requesting work.
- Ethical considerations and trust in model from humans.

It is clear that the size of the training data is insufficient. Deep learning models usually work on much larger data sets but the company has to decide whether it would be a fruitful use of time to annotate lots of data. There are some third-party solutions that can help with this problem. Some cloud services offer to label data for a small price per item if instructions are provided. Alternatively, the company could ask help desk operatives to label emails as they arrive to the desk. The second issue with annotation is that there can be ambiguity around the exact phrase to annotate. Often the fault cannot be expressed in a single word. For example, a common issue is that lighting becomes faulty. There are multiple valid ways of expressing this in an email such as:

1. "One of the light bulbs in area A is flickering."
2. "One of the light bulbs in area A needs replacing."
3. "One of the light bulbs in area A will not turn on properly."

Each of the scenarios above is describing the same type of problem. In scenario 1, the obvious fault entity is the single word, flickering. In scenario 3, the entire phrase "will not turn on" is required to accurately describe the fault. Some similar ambiguity is caused by the asset entity.

The potential benefit of extracting the fault, asset and trade entities is to help classify the problem type and to assign the correct trade type to the job. The asset and fault graphs show positive correlation between scores and training set size so they will benefit from annotating more data. Even if the asset and fault could be extracted every time using the named entity model, a method to classify the problem would still be needed. For example, if the asset is "plug fuse" and the fault is "broken", we would still need to classify this as an electrical problem so that the job can be assigned to the correct trade team. A text classifier would be needed to perform this task.

In order to automate the process of raising help-desk tickets, the model would need to integrate with an email inbox. This has been demonstrated to be possible but a company approved solution would need to be implemented. At the time of writing, this is an unknown. The named entity model was trained on a subset of emails that were all requests for work. If the model integrated with an inbox, it would need to first filter some of the standard admin emails from the inbox. Again, this would be a text classification problem, almost like a spam filter to create a pipeline of emails for the model that are all requesting work.

To summarise, there is a lot of work to do before we can provide the solution that will give the company a model that can automatically raise help tickets and it may prove too much effort for the value that it delivers. Nevertheless, there have been some clear lessons learned from this use case:

1. Always assess the complexity of your solution and weigh it against the value it will create.
2. Short proof of concepts are a good way of testing the water to see if something may be possible before embarking on a huge use case.
3. Emergent architecture principle: when beginning development, think about the architecture of the solution first. In this case, it would have been useful to consider if it was possible to integrate with an email inbox first. This would have identified the issue of needing to triage the inbox to filter admin emails earlier.
4. Opening discussions with stakeholders, even in the early stages of a proof of concept can be useful.

Other opportunities within the facilities management company to use NLP have arisen since sharing the concept on company social media channels.

4 Use Case 3: EV Charging Point Distribution

4.1 Background & Data

People are becoming ever more concerned with the irreversible changes we are making to our planet due to the rapid unsustainable consumption of fossil fuels and natural resources. As a result of this, having and owning a battery electric vehicle (BEV) is becoming increasingly popular as they are seen to be more sustainable. This change has meant that electric vehicle manufacturer Tesla has become the most valuable car company in the world in 2020 saying they aim to sell 500,000 cars in 2020.[21] As a result of this surge in popularity the issue of the placement of chargers around towns and cities has arisen. Managing the effect on the grid, the cost of building charging points, keeping range anxiety of drivers low are all factors to consider before placing chargers.[22] and[23]

The facilities management company wants to reduce their carbon footprint and proposes to convert their current diesel fleet of maintenance vehicles to electric vehicles in the next few years. The company offers staff generous home charging policies with free installation and remuneration for electricity costs. One simple solution would be to install home chargers for everyone and have them charge the vans at home overnight. Unfortunately, this is not feasible for a number of reasons. For example, a small number of vans regularly exceed the expected range of an electric van in a day and hence could not rely solely on home charging and some homes have no garage or driveway parking available. Furthermore, even if a home charger is feasible, it cannot be made compulsory, and many of the staff do not want the installation at their home. Therefore, a key challenge for the company is the effective placement of a private charging infrastructure.

This case study uses historic telemetry data from the current fleet of diesel vehicles to aid the task of choosing where to place the chargers. Telemetry data is gathered by an external system which collects data from "black boxes" attached to the vans. The data contains information on each separate journey that the vehicles make, including the start and stop address, as well as a breakdown of the moving, idling and stopped time. To visualise the data and understand how these addresses relate to our sites, the addresses have been geocoded to give the GPS coordinates. The company is interested in placing private chargers on the sites that they manage and so each of the van stops were spatially joined with the company sites so that, each time a van stopped in the data, the nearest company site was recorded. This allows us to understand the amount of traffic at each site which will be useful information when deciding where to place chargers. Figures 10 and 11 show the original telemetry data and the geocoded data with the spatial joins to the nearest company managed site.

Figures 10 and 11 about here.

The aim of the use case is to minimise the costs associated with transitioning the fleet to electric vehicles. The main criteria considered are the minimisation of the initial installation costs as well as the maximisation of engineer utilisation. Engineer utilisation is defined as the percentage of time spent doing productive work which adds value to the company. Performing maintenance or fixing faults is considered value-adding time and driving the vans or refuelling would be an example of non productive time. There is a natural trade-off between these two criteria, as increasing the number of charge points (hence increasing costs) will increase utilisation of the engineer due to less queuing at charge points and less need for diversions. There is some existing literature addressing similar problems. Fredriksson[24] discusses solving a pruned integer problem to select the optimal location of charging stations in a public charging network. Two other notable papers[23][22] published in this area of research are relevant to the problem. Lam 2014[22] provided a base to build on. The model was coded in Python defining the problem as in the paper. The next subsection defines the model and the rationale of the approach.

4.2 Methodology

This charging point placement problem is a multi-objective optimisation problem where the objectives are to minimise the cost of installation and maximise the utilisation of the engineers. Similarly to the job scheduling problem, the charging point placement problem can be solved using exact

methods, which may be more computationally expensive, or by using heuristic methods which find an approximate solution but can be much faster for larger problems. This use case is ongoing and, currently, the problem is being solved using a mixed integer linear program (MILP) coded in Python to find an exact solution. Based on experimentation, the computational cost of the exact methods were reasonable and therefore this approach was chosen over the approximate approach. Below, the model is defined as it currently is.

Model Inputs

These are the inputs that are required for the model:

1. **Number of Potential Charging Sites:** This is denoted by N. For the first attempt, a subset of sites where $N = 100$ was used.

2. **Charger Wattage:** Let p be the number of different charging point types we are considering. W is defined as a $1 \times p$ array denoting the different power of each type.

3. **Charge Ratio Matrix:** The charge ratio matrix is defined as R, a $1 \times p$ array where R_j is the time taken in minutes for a charger with wattage W_j to fully charge a vehicle.

4. **Installation Cost Matrix:** The costs matrix is defined by C, an $N \times p$ matrix where p is the number of different charger types we are considering. C_{ij} is the cost of installing a charger with wattage W_j at the site with index i. The company's EV Solutions team estimate the cost of a 50kW charger as £30k and the cost of a 7kW charger to be £2.5k.

5. **Demand:** F is the demand matrix with dimensions $1 \times N$ and F_i is the charging demand at site with index i. The charging demand at a given site is measured by the mean number of minutes per week spent at that site by all vehicles in the telemetry data.

6. **Mean Stop Time:** The mean stop time array defined as M, is a $1 \times N$ array where M_i is the mean amount of time that vans have stopped at the site with index i.

Objectives & Constraints

This optimisation model has multiple objectives: to minimise cost and maximise engineer utilisation. Engineer utilisation is difficult to define in the model so instead of maximising engineer utilisation, the model will minimise idle time and minimise unsatisfied demand. Unsatisfied demand is defined as the total demand at sites where there are no charge points. Minimising unsatisfied demand will encourage chargers to be placed at more sites. In this context, idle time at a particular site is defined as zero if there is no charger at that site. If there is a charger at the site, it is defined as the maximum of zero, and the time taken to do a full charge minus the average job time at that site. If there is a slow charger at sites where engineers tend to make quick visits, the idle time is high. Minimising idle time as an objective will encourage the use of faster chargers where vehicles are making short visits. Both concepts are defined mathematically later.

Hence, in total, there are three objectives: minimise installation cost, minimise idle time, minimise unsatisfied demand. Below, the trade-offs between these variables are listed to give an understanding of the interactions of the three objectives.

- **Cost vs Idle Time:** To minimise idle time, the model will favour installing 50kW chargers over 7kW chargers as they can provide charge much faster. Minimising idle time is desirable because it can be considered to be extra time that people spend charging after their job has finished. This is non-value time to the company.The 50kW chargers are more expensive, so the cost objective prefers the 7kW chargers.

- **Cost vs Unsatisfied Demand:** To minimise unsatisfied demand, the model will aim to install chargers in as many locations as possible, prioritising the busiest sites. Minimising unsatisfied demand is desirable because unsatisfied demand represents situations where people may need to divert from their scheduled work to charge the vehicle. This is non-value time to the company. Again, the cost objective will pull in the opposite direction, aiming to install fewer chargers.

There are two key decision variables in the model to introduce which will help make sense of the objective functions. The variable cs_i is a binary variable that indicates the presence of a charging point, or lack thereof, at site i. Another binary variable, x_{ij}, indicates the presence of a type j charge point at site i, differentiating between 7kW and 50kW charge points in this case. More explicitly, the variables are introduced below:

$$cs_i \in \{0,1\} \forall i \in \{0,...,N\}$$
$$x_{ij} \in \{0,1\} \forall i \in \{0,...,N\}, \forall j \in \{W\}$$

The three objectives can now be defined as follows:

$$minimise \sum_{i \in \{0,...,N\}} \sum_{j \in \{0,..,p\}} C_{ij} \times x_{ij} \tag{11}$$

$$minimise \sum_{i \in \{0,...,N\}} (1 - cs_i) \times F_i \tag{12}$$

$$minimise \sum_{i \in \{0,...,N\}} max\left(0, \left(\sum_{j \in \{0,..,p\}} R_j \times x_{ij}\right) - (M_i \times cs_i)\right) \tag{13}$$

The objectives are minimised according to the following constraints:

$$cs_i = \sum_{j \in \{0,..,p\}} x_{ij}, \forall i \in \{0,...,N\} \tag{14}$$

$$pw_i = \sum_{j \in \{0,..,p\}} x_{ij} \times W_j \forall i \in \{0,...,N\} \tag{15}$$

$$\sum_{i \in \{0,...,N\}} pw_i > \frac{E}{8} \tag{16}$$

$$\sum_{i \in \{0,...,N\}} pw_i < \frac{E}{3} \tag{17}$$

Equation 14 ensures that a maximum of one socket is installed at each potential location. Equation 15 defines a simplification variable which represents the power available at each site. The other constraints, 16 and 17, represent physical requirements. One key practical constraint is that the chosen infrastructure should be capable of delivering at least the minimum amount of energy required to meet the charging requirements of the vans. The amount of power required is calculated in the following way:

Let m be the mean number of miles travelled per day for a single van and n be the number of vans to transition from diesel to electric. The average total miles travelled per day for the whole fleet is, $M_{average} = m \times n$. The network should provide enough capacity to charge $M_{average}$ miles per day.

An assessment of the validity of the advertised vehicle range was conducted, and it concluded that adding one mile of vehicle range takes ≈ 0.54kWh of energy. Hence, the total energy capacity in the network should be $0.54 \times M_{average}$ kWh. Denote this energy by E. The required power to deliver this energy, P, measured in kW, is dependent on how well utilised the chargers are. If they were to be utilised for a full work day of 8 hours, then the minimum required power would be $\frac{E}{8}$. Discussions with subject matter experts indicated that the chargers should be used for at least an average of 3 hours per day to have a good return on investment. Hence, there is a constraint that the total power in the network of chargers should be between $\frac{E}{8}$ and $\frac{E}{3}$.

In this situation, there are multiple objectives, measured in different units. One possible approach would be to scale the individual objectives and define the main objective as a linear combination of the scaled objectives. Instead, a pareto-front optimisation is performed using the

epsilon-constraint method. Pareto-front optimisation aims to find the set of pareto efficient solutions. A pareto efficient solution is one where any of the individual criterion or objectives cannot be improved without detriment to one or more of the others. The set of solutions is often visualised graphically and described as the pareto front. Providing these set of solutions to a decision maker can help them make focused trade-offs without being concerned about all the parameters.

There are multiple approaches to find the pareto front and, for this use case, the epsilon-constraint method is used. It is not the intention of this paper to provide an overview of the entire method but to demonstrate how it was applied. This paper[25] provides the details for those interested. In simple terms, the multi-objective model is converted to a single objective model by considering one of the objectives to be the main objective function. The remaining objectives are reformulated to be new constraints. To find the pareto front, the model is solved repeatedly with the restrictions on the new constraints changing slightly each time so that the solution space is explored systematically. In this application of the method, the installation cost objective is treated as the main objective with the unsatisfied demand and idle time objectives being converted to constraints.

The model objective and extra constraints are formulated below:

$$minimise \quad f_p + 10^{-5} \times (f_1 + f_2) \tag{18}$$

where:

$$f_p = \sum_{i \in \{0,\ldots,N\}} \sum_{j \in \{0,..,p\}} C_{ij} \times x_{ij}$$

$$f_1 = \sum_{i \in \{0,\ldots,N\}} (1 - cs_i) \times F_i$$

$$f_2 = \sum_{i \in \{0,\ldots,N\}} max\left(0, \left(\sum_{j \in \{0,..,p\}} R_j \times x_{ij}\right) - (M_i \times cs_i)\right)$$

subject to:

$$f_1 \leq \epsilon_1 \tag{19}$$

$$f_2 \leq \epsilon_2 \tag{20}$$

The new objective function is the primary objective, installation cost, plus the other objectives with a very small coefficient to break tie in the case of two solutions of equal cost. Systematically changing the values of ϵ in each constraint and iterative solving of the objective function provides us with a set of pareto optimal solutions. In this case, 20 values of each epsilon were used, equally spaced between the minimum and maximum values of the respective functions. Note that these epsilon constraints are in addition to the original constraints.

Solver choice and Output

The model has been coded in Python using gurobi optimisation software.[26] Other viable options for the optimisation software are cplex, which is IBM's solver and the pyscipopt package which is a Python interface for using the scip optimisation software. Each have slightly different licensing policies and performance. The decision to choose gurobi was based on both performance and licensing.

The output of the solver is the set of pareto-efficient solutions. In this case, there are three objectives and so it is possible to plot the pareto front in 3-dimensional space.

Figure 12 about here.

Figure 12 shows the pareto front from two different angles. Each point on the pareto front is a solution which contains a number of charging locations and the type of chargers recommended at each location. The colour of the solution represents the proportion of 50kW chargers in the

solution. 50kW chargers are a lot more expensive, but provide charge much faster. Therefore the solutions which contain lots of these types of chargers are high on the cost axis and low on the idle time axis. Also, since we need to place fewer chargers in these types of solutions, they tend to be high on the unsatisfied demand axis. Practically speaking, this means that solutions with high proportion of 50kW chargers will require engineers to deviate from their schedule more often than those with high proportions of 7kW chargers. However, they will spend much less time charging.

4.3 Results & Model Validation

Since the model provides multiple potential solutions to the problem, a decision needs to be made on which solution to choose. The approach in this case involved discussing the practical feasibility of placing chargers at certain sites. For some sites, there is no electrical capacity to install a charge point and in others there is not the space for one. This helped us to rule out some of the solutions in the pareto front. Unfortunately, survey information for all the sites does not exist so it was not possible to constrain the model prior to solving it. Discussions around the potential budget for installing chargers ruled out other solutions. Again, this information could have been used as inputs to the model, however the contact was not clear on this before the solutions were presented. Simulations were run to help understand the impact of samples of the remaining feasible solutions on the business. The solution with the most agreeable impact was chosen as the final model. A sample of 5 solutions were tested and the process of sampling was semi-random, choosing solutions from different areas of the pareto front. In hindsight, a more methodical approach may have been to choose solutions based on some distance function. For example, Euclidean distance from the origin.

For each solution sample, simulations of engineer journeys are run, under the assumption that the vans are electric vehicles with possible charging point locations and types determined by the solution. The simulation collects some statistics of interest so that the likely impacts of installing charging infrastructure based on the model recommendations can be understood. The simulation is built using simpy,[27] a process-based, discrete-event simulation framework for Python. Simpy makes it easy to model shared resources allowing useful metrics such as wait time distributions and charger utilisation can be collected. The simulation works by initialising a number of van objects with attributes such as current range and miles travelled.

The vans travel between 100 sites, stopping to do work and charging when necessary. The spatial relationship between the sites is captured in a distance matrix and the time taken to travel between each site is captured in a journey-time matrix. The distance matrix and time matrix are obtained from the GPS coordinates of the sites and by using the open-route service API.[28]

At each stage, the van chooses its next journey by using a transition matrix which is calculated empirically from the telemetry data. The most commonly visited 100 sites in the data were extracted and a zero matrix of dimensions 100x100 was initialised. For each journey between two of these sites in the data, the value in the corresponding place in the matrix was increased by one. This ensured that the frequency of visits to each site in the simulation was representative of reality and that the journey lengths were more consistent. For example, engineers are more likely to choose a site close to them for their next job.

A job time is then simulated to determine how long the van will wait at its next job. This is important as the amount of stopped time directly affects the amount of journeys the van will do in a day and hence the amount of charging the van will need to do. The stopped times are simulated from a distribution that is elicited from the data. The distribution of stopping times can be observed by creating a density plot of the "Stopped" column in the telemetry data. The goal is to find a distribution that closely matches the observed data and to simulate stop times from that distribution. MATCH elicitation tool[29] offers different methods for eliciting a distribution. The roulette method was used to find a truncated log-normal distribution which closely matches that of the stopping times. Figure 13 is a graph showing the density plot of the actual data, and a density plot of some simulated data from a truncated log-normal distribution.

Figure 13 about here.

Figure 14 is a high level process map that shows the inputs and outputs from the validation.

Figures 14 and 15 about here.

Figure 16 shows some of the key outputs from the solution delivered to the contract that the model was trialled with. The average charger utilisation was 31.5% of the working day of 9 hours in the simulation. This equates to approximately 3 hours of usage per charger per day. In the simulation the vans travelled an average of 35.6 miles per day which is equivalent to the data that is seen in reality. This is important as it confirms that the amount of charging that is recorded in the simulation is the amount that can be expected in reality. The amount of charging time, in mean number of minutes per day, per van is 22.5. According to the wait time distributions, the mean amount of waiting time at a charger per day is 7 minutes. Hence the mean impact on an engineers day is close to 30 minutes per day which can be viewed as a 6.3% reduction in engineer utilisation. This information is important to operational managers so they can understand the potential impact on their service.

Figure 17 shows the recommended locations of the chargers on a map.

Figure 16 about here.

Figure 17 about here.

4.4 Conclusions

At the time of writing, the company is looking to transition many of their diesel fleet to electric. It has set itself ambitious targets of removing diesel vehicles from the fleet by 2025 and that target is currently at risk of not being met. This appears largely to be due to risk averse operational decision makers who are cautious of the uncertainty around transitioning their fleet and of the risk to their operations. This use case aims to help provide some clarity and reduce uncertainty for operational leads around the impact it will have on their operations whilst also providing a method and a road map for how and where to start transitioning. So far, the absence of any clear method has proved to be a barrier to achieving the company's target and the hope is that one or two successful use cases will trigger a rapid uptake in electric vehicles with the company.

The model outputs and simulations have been presented back to one contract manager who has agreed to take a more staged approach to transition by installing some of the recommended infrastructure and transitioning a portion of their fleet. Future work on this use case might be to provide a more detailed road map including how best to stage the transition and thereby compartmentalise the risk. At the time of writing, work with a second contract is underway and interest from other international business units is developing. One of the difficulties faced in implementing this model has been the interpretability of the model- it can be difficult explaining some of the methods and outputs to non-technical audiences. Also, the commercial aspect of the use case requires some careful navigation. It may be the case that the infrastructure is installed and managed by another part of the business and the contractor is recharged for the electricity that it uses. In this case, the utilisation of the charge points would be key to the return on investment. It is unclear at the moment where this use case will lead, but the hope is that the model could play a part in the central EV Solutions teams business model for transitioning not only the company's own fleet, but also some of its customers' fleets.

5 Discussion

Data analytics is a growing field across all organisations with companies keen to create dedicated sections dealing with innovative applications of data science. Interventions require a combination of IT skills, application of statistical analysis and machine learning techniques and business know-how. These skills are not readily available and companies are reluctant to invest resources into data science unless they are confident that it will prove beneficial in their unique business. Via a part-funded KTP, we have demonstrated that considerable, valuable insight can be obtained from applying data science techniques to facilities management. In particular, techniques including optimisation, deep learning, and simulation have been used in three different but related use cases.

Opportunities for realising benefits from data analytics arise in two ways. Companies keep abreast of changes in the business environment and are mindful of maintaining and enhancing their competitive position by embracing emerging methodologies. There is a willingness to investigate new methods. In addition, the clients of service company's have ongoing challenges that they need to overcome to improve their business and can read in their trade press what other people are trying. This can be in terms of new services, such as automating customer calls, or improving existing services, such as streamlining job scheduling. There are also overarching business imperatives, the need to cut the carbon footprint being most important in a global company, hence the enthusiasm for electric vehicles and the interest in appraising the entry point and needs for a new way of working.

As shown in the use cases, data quality, business domain knowledge and team working are key to success. There are considerable collateral benefits from data analytics interventions as improvement in data capture in itself can focus attention on the processes and lead to improvement from a so-called Hawthorne effect. Improving data capture whenever possible will, in turn, always improve the quality of the analyses. Historic data were needed for the job scheduling use case and it was an interesting experience liaising with staff to assemble this in a suitable form. Data from different sources, such as GPS and odometers had to be combined with engineer supplied times when jobs start and finish. Data harmonisation was sometimes labour-intensive when needing to take input from people working in different parts of the organisation and build a whole ontology of potential variables and their possible relationships. However, once completed, this harmonisation contributed to the ease of implementing and further improving the new methods.

Data analytics is important but is not the only consideration. Results then have to be presented in an accessible form for the user, consumer and customer. This step is non-trivial and meaningful clear data visualisation and user interaction have been made as appealing and flexible as possible to enhance the buy-in from staff. Due care was taken with change management and the treatment of staff familiar with well-established working methods and now facing a new way of working.

It was very important that the new data science approach, including use of a wide range of tools, was embedded in the company and supported. A coding club was created and short, voluntary lunchtime sessions on Python and data analytics methods were held and were very popular with staff. Online training material was especially designed for staff and they were encouraged to educate themselves via other online courses and continuing professional development identified during the research. The KTP embraced the services of an apprentice in the company and a number of student projects were set up to build a body of experience. This led to a new data analytics focused unit within the company which will ensure the legacy of the KTP.

The business required a sound return on investment from data analytics and this was shown from (1) maximising the productivity of engineers, (2) reducing the time taken to process help desk emails, (3) creating a data driven method to provide objective decision support for charger placement.

The lessons learned from this work and the management implications are as follows:

- Ensure careful scoping of the project and collaboration/agreement between stakeholders. This eases the gathering and integration of data.
- Be sure to define success and the mechanism for measurement of progress. The helps to keep the project on track.
- Use visualisation and clear presentation of progress road maps, results and implications giving feedback to stakeholders. This promotes the adoption of new methods.

The novelty of the research is the demonstrated applications of data analytics in facilities management and the outcomes have justified further investment in this area.

6 Acknowledgements

The research in this manuscript were made possible by a Knowledge Transfer Partnership (KTP) between Engie, Innovate UK, and Newcastle University's School of Mathematics, Statistics and Physics and the School of Computing. The project received funding from Innovate UK, Knowledge Transfer Partnership number 11315.

Map data copyrighted OpenStreetMap contributors and available from https://www.openstreetmap.org.

References

[1] S. Coleman, "Data excellence in smes through engagement in university partnerships," in *Big Data in Small Business: Data-driven Growth in Small and Medium Sized Enterprises* (C. Pedersen, A. Lindgreen, T. Ritter, T. Ringberg, and E. Elgar, eds.), 2021 in press.

[2] J. King and C. Perry, *Smart buildings: Using smart technology to save energy in existing buildings*. American Council for an Energy-Efficient Economy, 2017.

[3] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ." https://www.openstreetmap.org , 2021.

[4] A. Ekici and A. Retharekar, "Multiple agents maximum collection problem with time dependent rewards," *Computers Industrial Engineering*, vol. 64, no. 4, pp. 1009–1018, 2013.

[5] S. E. Butt and T. M. Cavalier, "A heuristic for the multiple tour maximum collection problem," *Computers Operations Research*, vol. 21, no. 1, pp. 101–111, 1994.

[6] K. Tan, L. Lee, Q. Zhu, and K. Ou, "Heuristic methods for vehicle routing problem with time windows," *Artificial Intelligence in Engineering*, vol. 15, no. 3, pp. 281–295, 2001.

[7] N. A. El-Sherbeny, "Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods," *Journal of King Saud University - Science*, vol. 22, no. 3, pp. 123 – 131, 2010.

[8] J. K. Lenstra and A. H. G. R. Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.

[9] J. Forrest, S. Vigerske, H. G. Santos, T. Ralphs, L. Hafer, B. Kristjansson, J. P. Fasano, E. Straver, M. Lubin, R. Lougee, J. P. Goncal, H. I. Gassmann, and M. Saltzman, "coin-or/cbc: Version 2.10.5," Mar. 2020.

[10] L. Perron and V. Furnon, "Or-tools." https://developers.google.com/optimization/.

[11] D. Montgomery, *Introduction to statistical quality control*. New York, NY [u.a.]: Wiley, 3. ed ed., 1997.

[12] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[14] D. Mimno, H. Wallach, E. Talley, M. Leenders, and A. McCallum, "Optimizing semantic coherence in topic models," in *Proceedings of the 2011 conference on empirical methods in natural language processing*, pp. 262–272, 2011.

[15] R. Jiang, R. E. Banchs, and H. Li, "Evaluating and combining name entity recognition systems," in *Proceedings of the Sixth Named Entity Workshop*, pp. 21–27, 2016.

[16] SpaCy, "Spacy api architecture documentation," September 2020. https://spacy.io/models.

[17] R. Weischedel, E. Hovy, M. Marcus, M. Palmer, R. Belvin, S. Pradhan, L. Ramshaw, and N. Xue, "Ontonotes: A large training corpus for enhanced processing." Report LDC2013T19, Linguistic Data Consortium, 2013. https://catalog.ldc.upenn.edu/LDC2013T19.

[18] SpaCy, "Spacy ner annotation tool," September 2020. (Accessed on 09/01/2020).

[19] G. Crichton, S. Pyysalo, B. Chiu, and A. Korhonen, "A neural network multi-task learning approach to biomedical named entity recognition," *BMC bioinformatics*, vol. 18, no. 1, p. 368, 2017.

[20] M. W. Browne, "Cross-validation methods," *Journal of Mathematical Psychology*, vol. 44, no. 1, pp. 108–132, 2000.

[21] BBC News, "Tesla overtakes toyota to become world's most valuable carmaker - bbc news." https://www.bbc.co.uk/news/business-53257933: :text=Tesla

[22] A. Y. S. Lam, Y. Leung, and X. Chu, "Electric vehicle charging station placement: Formulation, complexity, and solutions," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2846–2856, 2014.

[23] A. Y. Lam, Y.-W. Leung, and X. Chu, "Electric vehicle charging station placement," in *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, IEEE, oct 2013.

[24] H. Fredriksson, M. Dahl, and J. Holmgren, "Optimal placement of charging stations for electric vehicles in large-scale transportation networks," *Procedia Computer Science*, vol. 160, pp. 77 – 84, 2019. The 10th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN-2019) / The 9th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2019) / Affiliated Workshops.

[25] Z. Fan, H. Li, Caimin Wei, W. Li, Han Huang, X. Cai, and Z. Cai, "An improved epsilon constraint handling method embedded in moea/d for constrained multi-objective optimization problems," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 1–8, 2016.

[26] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2020. http://www.gurobi.com.

[27] Stefan Scherfke, Ontje Lünsdorf, "Simpy: Simulation framework," July 2020. https://simpy.readthedocs.io/en/latest/index.html.

[28] GIScience, "openrouteservice." https://github.com/GIScience/openrouteservice/, 2021.

[29] D. Morris, E. Oakley, and J. Crowe, "A web-based tool for eliciting probability distributions from experts, environmental modelling software," vol. 52, pp. 1–4, 2016.

Author Biography

Daniel Walker: Daniel Walker received an MSc in Mathematics and Statistics from Newcastle University in 2018. He has recently completed a two-year Knowledge Transfer Partnership (KTP) as an associate, where he worked closely with the company and Newcastle University, leading on data science projects to add value to the company through the application of academic methods. He engages with academic communities, hosting webinars or presenting to the European Network for Business and Industrial Statistics (ENBIS), Analyst Network North East (ANNE) and the Royal Statistical Society (RSS), as well as other business communities. Throughout the KTP, he has up-skilled in technical and managerial areas by taking courses in machine learning, SQL, Six-Sigma and chartered management.

Author Biography

Martin Ruane: Martin Ruane has an MBA, holds a full professional qualification for the Institute of Ratings, Revenues and Valuation and is a Black Belt in Lean Six Sigma. He currently works for the company as Programme Director where he leads on business improvement activities. In this role, he has pioneered work in the fields of robotic process automation and data science. Martin is a regular speaker at international conferences on digital improvements, data science and robotic process automation.

Author Biography

Jaume Bacardit: Jaume Bacardit has received a BEng, MEng in Computer Engineering and a PhD in Computer Science from Ramon Llull University, Spain in 1998, 2000 and 2004, respectively. He is currently Reader in Machine Learning at Newcastle University in the UK. Bacardit's research interests include the development of machine learning methods for largescale problems, the design of techniques to extract knowledge and improve the interpretability of machine learning algorithms and the application of these methods to a broad range of problems, mostly in biomedical domains.

Author Biography

Shirley Coleman: Shirley Coleman has a BSc in Maths, MSc in Statistics and PhD in statistics, computer graphics and morphometry from Newcastle University where she is Technical Director of the Industrial Statistics Research Unit. She works on data analytics in Small and Medium Enterprises (SMEs) and specialises in statistical and machine learning techniques applied to company data. She publishes in trade and academic journals and is co-editor of several books. She is a past President of the European Network for Business and Industrial Statistics (ENBIS) and a Chartered Statistician of the Royal Statistical Society, instrumental in mentoring early career statisticians and developing relationships with business and industry.

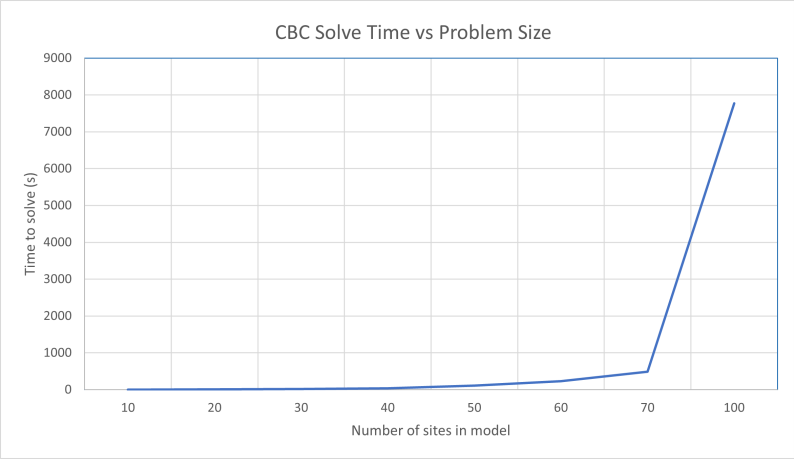

Figure 1: Solve times in seconds for CBC solver with different problem sizes

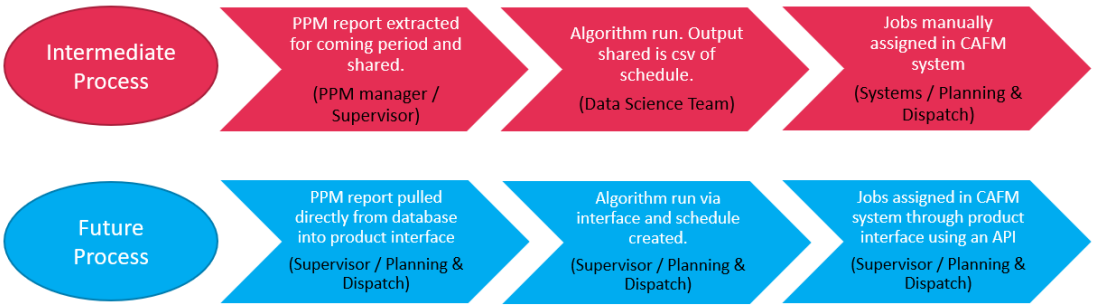

Figure 2: Intermediate process and future process (high level)

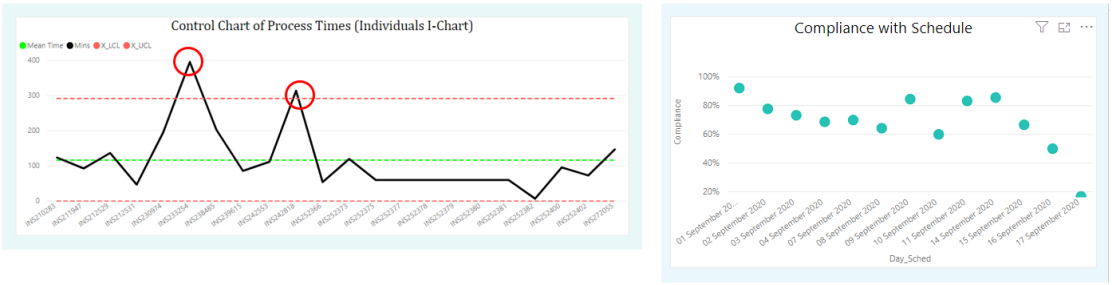

Figure 3: Control chart of gas boiler servicing times (left) & compliance with schedule (right)

Metric	Result / Benefit
Total travelling time	Comparisons with 6 months of baseline data show that the total amount of travelling time was reduced by 22%
Process times for jobs	As a side-effect possibly due to previous over-reporting the average job times have reduced for many job types, some by up to 43%
Customer experience	Scheduling jobs ahead of time has allowed for trialling an appointment system. This gives the customer more notice and mitigates potential access problems when attending site.
Availability of data	PowerBI dashboards now present a wealth of operations data to the supervisors. This should help them improve their performance management - an area which was identified as lacking by management.
Process variation	With access to the data, anomalies can be easily identified and investigated. Where issues are spotted they can be resolved, removing a potential cause of future variation.

Table 2: Results and benefits delivered by this use case

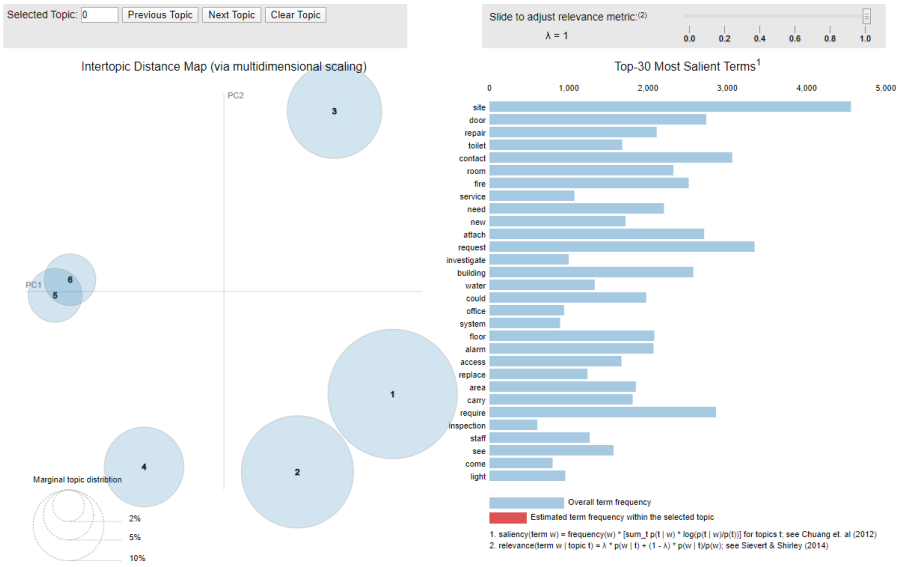

Figure 4: Latent Dirichlet Allocation of email bodies from an FM help desk

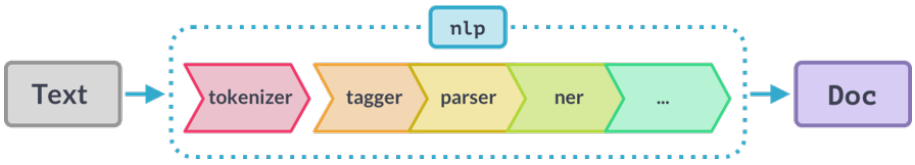

Figure 5: spaCy's natural language processing pipeline.[16]

Original Email Body	Ground Truth Site	Ground Truth Job Type	Annotated Entities
Hi, there is a burst pipe at abc school	ABC Primary School	Plumbing - Pipes	{"Site":[29,38], "Asset":[21:24], "Fault":[15:19]}

Table 3: Example training data used to train a supervised learning model

TYPE	DESCRIPTION
PERSON	People, including fictional.
NORP	Nationalities or religious or political groups.
FAC	Buildings, airports, highways, bridges, etc.
ORG	Companies, agencies, institutions, etc.
GPE	Countries, cities, states.
LOC	Non-GPE locations, mountain ranges, bodies of water.
PRODUCT	Objects, vehicles, foods, etc. (Not services.)
EVENT	Named hurricanes, battles, wars, sports events, etc.
WORK_OF_ART	Titles of books, songs, etc.
LAW	Named documents made into laws.
LANGUAGE	Any named language.
DATE	Absolute or relative dates or periods.
TIME	Times smaller than a day.
PERCENT	Percentage, including "%".
MONEY	Monetary values, including unit.
QUANTITY	Measurements, as of weight or distance.
ORDINAL	"first", "second", etc.
CARDINAL	Numerals that do not fall under another type.

Figure 6: List of default entities in the spaCy NER model.[16]

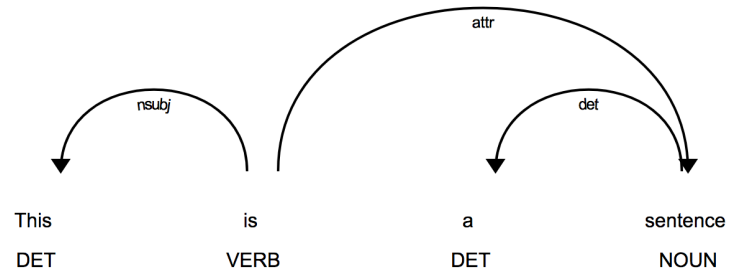

Figure 7: An example of dependency parsing from the official spaCy web page

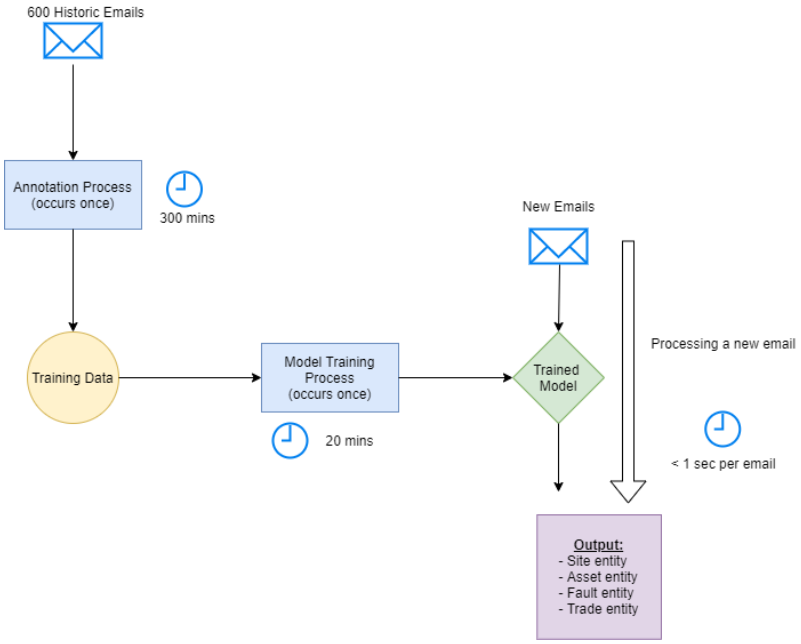

Figure 8: Flow chart of model training and use process

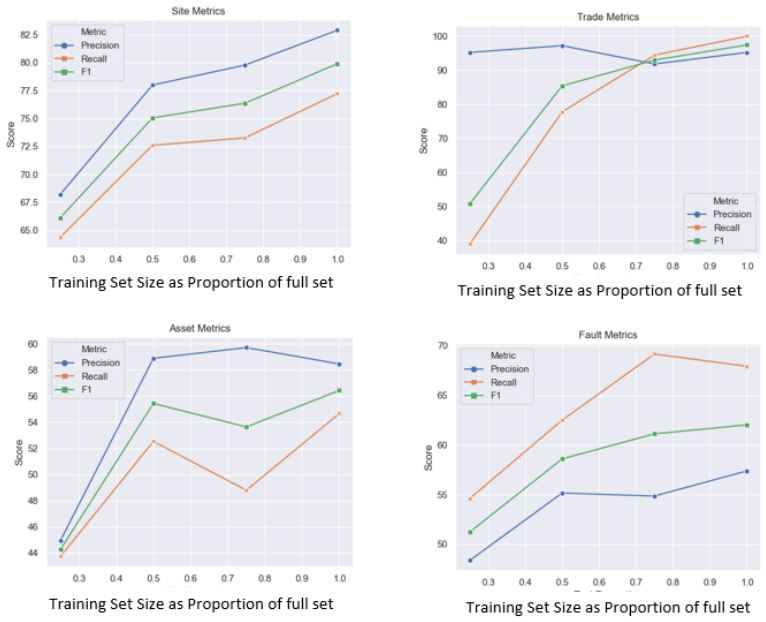

Figure 9: Mean precision, recall and f1 scores as a function of training set size

Vehicle	Start	End	Shift Time	Moving	Idling	Stopped	Start Location	Stop Location	Distance (mi)	Max (mph)	Avg (mph)
XXXXXXX	06/01/2020 12:53	06/01/2020 13:05	00:41:51	00:11:54	00:00:00	00:29:57	Redacted Address	Redacted Address	4.2	34.8	21.3
XXXXXXX	06/01/2020 13:35	06/01/2020 14:05	00:34:29	00:30:30	00:00:00	00:03:59	Redacted Address	Redacted Address	9.6	42.3	18.9
XXXXXXX	06/01/2020 14:09	06/01/2020 14:25	00:26:03	00:15:36	00:00:00	00:10:27	Redacted Address	Redacted Address	4.5	46	17.2
XXXXXXX	06/01/2020 14:35	06/01/2020 14:51	00:20:48	00:12:41	00:03:05	00:05:02	Redacted Address	Redacted Address	4.3	45.4	16.6
XXXXXXX	06/01/2020 14:56	06/01/2020 14:59	00:20:09	00:00:00	00:02:26	00:17:43	Redacted Address	Redacted Address	0	0	0

Figure 10: Raw telemetry data with redacted locations

Vehicle	Start	End	Shift Time	Moving	Idling	Stopped	Start Location	Stop Location	Distance (mi)	Max (mph)	Avg (mph)	Closest_Site	Distance_to_Closest_Site (km)
XXXXXXX	06/01/2020 09:58	06/01/2020 09:58	00:02:30	00:00:21	00:00:00	00:02:09	Redacted Address	Redacted Address	0	0	0	Redacted Site	0.019769183
XXXXXXX	06/01/2020 10:01	06/01/2020 10:04	00:30:40	00:03:31	00:00:00	00:27:09	Redacted Address	Redacted Address	0	6.2	0	Redacted Site	0.020889696
XXXXXXX	06/01/2020 10:31	06/01/2020 10:50	00:26:38	00:16:58	00:02:03	00:07:37	Redacted Address	Redacted Address	5	34.2	15.7	Redacted Site	0.098075473
XXXXXXX	06/01/2020 10:58	06/01/2020 11:12	00:15:27	00:13:44	00:00:00	00:01:43	Redacted Address	Redacted Address	5	35.4	21.7	Redacted Site	0.017449555
XXXXXXX	06/01/2020 11:13	06/01/2020 11:14	00:13:16	00:01:10	00:00:00	00:12:06	Redacted Address	Redacted Address	0	3.1	0	Redacted Site	0.029026961
XXXXXXX	06/01/2020 11:27	06/01/2020 11:28	00:19:56	00:01:18	00:00:00	00:18:38	Redacted Address	Redacted Address	0	3.1	0	Redacted Site	0.068597842

Figure 11: Geocoded telemetry data with redacted locations

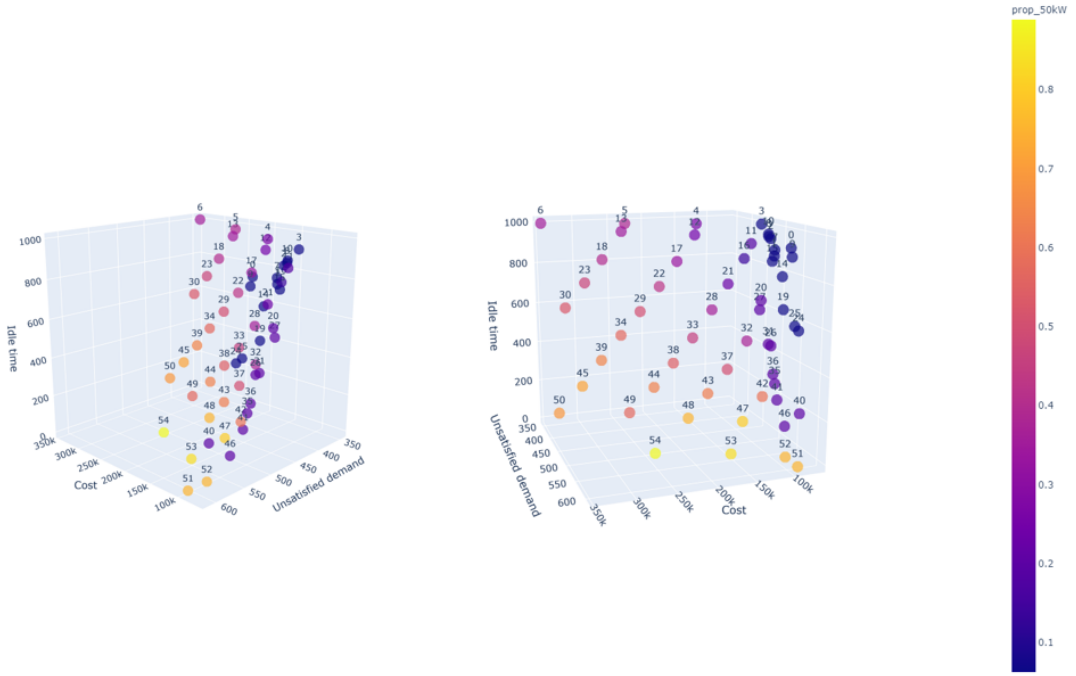

Figure 12: Set of pareto-efficient solutions coloured by proportion of 50kW chargers in solution

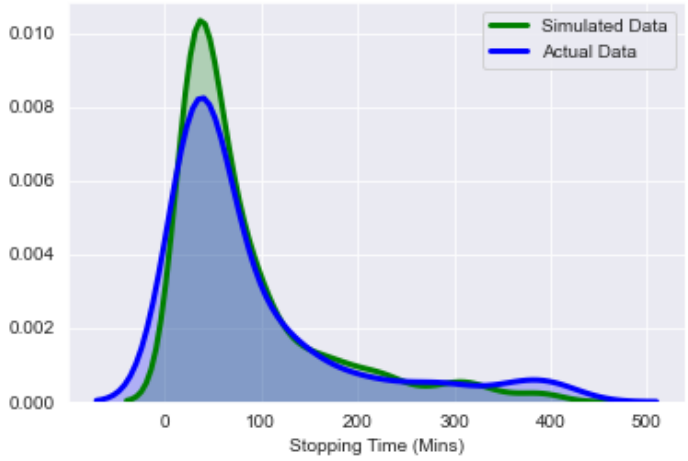

Figure 13: Density plot of real and simulated data.

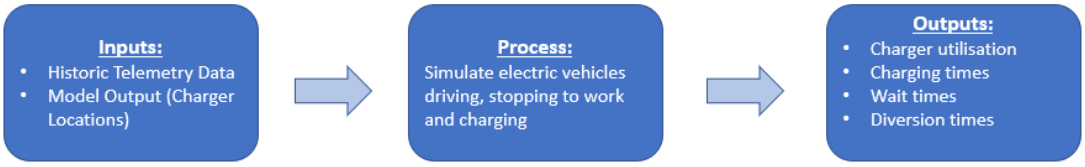

Figure 14: High level validation process map.

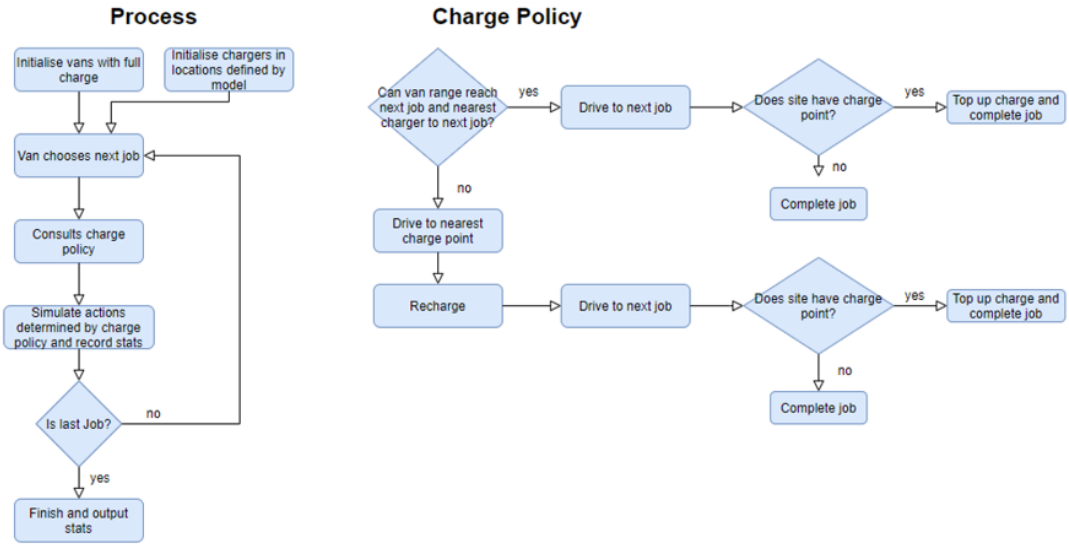

Figure 15: Detailed map of simulation process

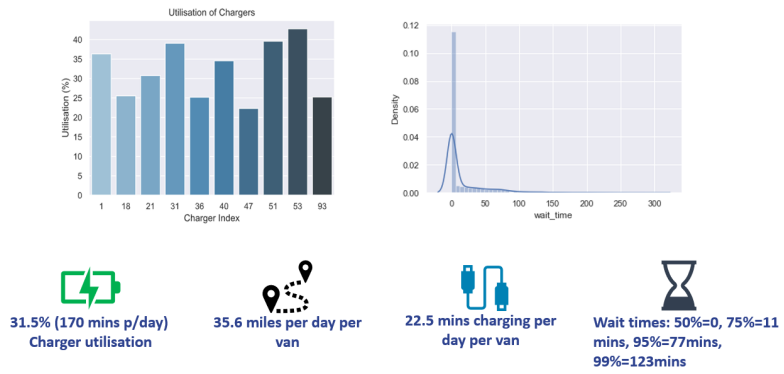

Figure 16: Outputs from the simulation.

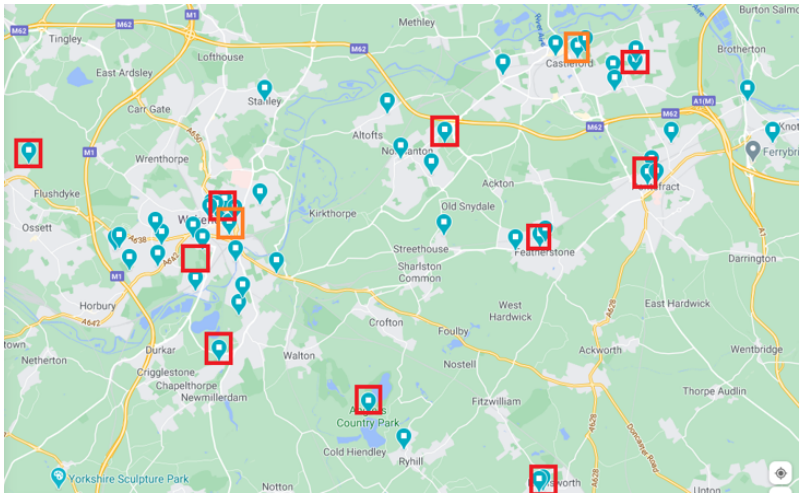

Figure 17: Visualisation of charger locations