

Title of Dissertation

Author’s Name

June 8, 2007

MSc in High Performance Computing

The University of Edinburgh

Year of Presentation: 2007

Abstract

This is the bit where you summarise what is in your dissertation.

Contents

[Chapter 1 Introduction 1](#_Toc169347395)

[Chapter 2 Background Theory 2](#_Toc169347396)

[2.1 The easy bits 2](#_Toc169347397)

[2.2 The more difficult bits 2](#_Toc169347398)

[2.2.1 Hard bits 3](#_Toc169347399)

[2.2.2 Even harder bits 3](#_Toc169347400)

[Chapter 3 Experimental design 5](#_Toc169347401)

[Chapter 4 Results and Analysis 6](#_Toc169347402)

[4.1 Some results 6](#_Toc169347403)

[4.1.1 More results 6](#_Toc169347404)

[Chapter 5 Conclusions 8](#_Toc169347405)

[Appendix A Stuff which is too detailed 9](#_Toc169347406)

[Appendix B Stuff which no-one will read 10](#_Toc169347407)

[References 11](#_Toc169347408)

List of Tables

[Table 1: This is a simple table. More complicated tables can have headings which pass over more than one column. 5](#_Toc169345869)

List of Figures

[**Figure 1: Workflow Diagram** 2](#_Toc522469817)

[**Figure 2: HTML source code with HTML style of one of ‘Fish & Chips’ shops’ websites** 6](#_Toc522469818)

[**Figure 3: HTML source code of one of ‘Fish & Chips’ shops’ websites** 6](#_Toc522469819)

[**Figure 4: geographical coordinate system with a cartesian coordinate systems** 7](#_Toc522469820)

[Figure 1: The University Crest 11](#_Toc522469821)

[Figure 2: Simple Gnuplot example 13](#_Toc522469822)

Acknowledgements

This template is based on David Henty’s LaTeX template, which in turn was based on a LaTeX template created by Prof. Charles Duncan for MSc students in the Dept. of Meteorology. His acknowledgement follows:

*This template has been produced with help from many former students who have shown different ways of doing things. Please make suggestions for further improvements.*

# Introduction

This chapter provides a general overview of the dissertation, including background, aims, methods, study value, and main structure of this research.

## Dissertation background

In the era of rapid development of the Internet and smartphones, searching restaurants or specific food has become easier. For example, with Google Maps, customers could find the type of restaurant or food they are interested in more simply and more accurately. At the same time, restaurant operators can use these techniques to better expand their business [1]. On discovering this business opportunity, an increasing number of restaurant operators are advertising the link to their menu on some information applications and websites, such as Google Maps and TripAdvisor [2]. This has resulted in a large amount of restaurant information such as menu and restaurant description, being publicly available in a digital form. In terms of ordinary diners, the content in restaurants’ websites can only provide them with a reference for catering. However, in the eyes of data workers, a lot of potential valuable information can be mined from these restaurants’ data. That means some valuable findings such as regional differences in a country or region can be found if using data mining techniques [3] on restaurants’ websites. This dissertation will introduce a method to get, process very messy restaurants’ websites data, mining regional features and classifying regional content from this data to reveal the regional differences.

## Research aim and research focus

The aim of this dissertation is to mine menu data from the “Fish&Chips” shops, revealing regional differences in the UK based on the geographical distribution of the content of the menu data. For example, 'Haggis' is a traditional food in Scotland and widely distributed, while rarely seen in England. According to the methodologies applied in this dissertation, we could provide evidence that 'Haggis' is loved by the Scottish people and it is a regional dish in Scotland. ‘Fish & Chips’ is one of the most famous foods in the UK and there are more than 1,000 ‘Fish & Chip’ shops in this country [4]. To achieve the project aim we will obtain the raw HTML data from the websites of some of these ‘Fish & Chip’ shops, and then employ data cleaning, mining, and visualisation techniques to find the content associated with particular regions.

## Research methods

In terms of data crawling, the dissertation will illustrate the selection of data sources and methods for crawling data from ‘Fish & Chip’ shops’ websites in the UK. The data cleaning procedure focuses on extracting and cleaning text content which is used for exploring regionality from the website HTML content, such as single independent words, noun phrases, and word pairs. The methods used for extracting and cleaning HTML content is the combination of Regular Expressions, HTMLParser and Natural Language Processing (NLP) (will be detailed in 2.2). Considering the data mining procedure of the research, data visualisation technologies will be applied to mine the regional features based on the geographical distribution of the extracted content. In terms of the classification (regional content and national content) of the extracted datasets, the project employs machine learning methods, such as decision trees and regression classifiers (will be detailed in 2.4) to generate the regionality result. Specifically, this research is an iterative process and includes four rounds of evaluation and improvement (showed by Fig. 1) since the entire study is an exploratory process. Excepting for the first iteration, each of the remaining iterations depends on the result of the last iteration. This means the project knows what to proceed next only after getting and evaluating the results of each iteration. Besides, there are no existing criteria to verify the rationality of the method selection and the correctness of the results. Thus, only after each iteration has finished, the project can know whether the choice of method is reasonable and whether the result is correct. In addition, the features that could be used for reflecting regionality of the text is unknown and the evaluation of regionality content is based on the evaluator’s experiences to some extent. For example, the evaluator knows the ‘irn bru’ is a Scottish drink, so when this phrase is judged to be regional, the evaluator can assume the decision is correct. Therefore, regional features and regional results are derived from each iteration, including exploration, evaluations, and improvements. In each iteration, the project may use or update the methods in the previous iteration. Besides, each iteration will also evaluate the results to identify problems and propose improvements for the next iteration.



**Figure 1: Workflow Diagram**

## Value of the research

The research is an innovative study that links seemingly unrelated menu information to regional differences of the UK through exploring regional content from the messy menu dataset. Thus, the biggest challenge for this project is that there are no existing ideas or methods to refer to and all ideas and methods require the project to explore and evaluate. Fortunately, the project succeeded in finding a solution to explore regional content and reflect regional differences, including ideas for finding regional features and methods for regional words classification. The methods and algorithms used in this project are universal, and they can also be used to find regional differences in other countries or used in similar studies. As a result, the project laid the foundation for similar subsequent researches.

## Structure of the dissertation

The structure of this dissertation is organised as follows: Chapter 2 covers background knowledge, which mainly illustrates the main techniques and algorithms used in this research. Chapter 3, Chapter 4, Chapter 5 and Chapter 6 introduce the details of each iteration in the project, including methodologies, findings, evaluations, and improvements and summary and future work. Chapter 3 describes the first iteration, presenting the procedures of exploring features of regional independent single words. Chapter 4 is related to the second iteration, which describes the application of decision tree to get regional results of the single word. This chapter uses two types of decision tree algorithms and the makes comparison between the two algorithms. Chapter 5 covers the third-round iteration, which introduces the logistic regression to obtain the probability that the single independent word is judged as a regional word. This chapter focuses on evaluating the importance of the selected features and the threshold regarding probability, in order to identify the number of probabilities exceeds the threshold, which would be judged as regional word. Chapter 6 is about the fourth iteration, which demonstrates the results of using the other two kinds of datasets (noun phrases and word pairs). Chapter 7 is a conclusion about this research. Chapter 8 provides a description of the improvement could be included in future work and also introduces the limitations, and recommendations.

# Background Theory

This chapter focuses on describing the techniques and methods used in the dissertation. Section 2.1 states the selection of data source and the definition of the web crawling. Section 2.2 presents an overview of data cleaning procedure and techniques used in the project. Section 2.3 provides the description of core algorithm and techniques used for geographic data visualisation. Section 2.4 aims to illustrate the data mining process in the project and some machine learning methods used for classifying regional content.

## Web crawling

Before crawling data from websites of ‘Fish & Chips’ shops, the dissertation selected data sources that included food delivery websites such as Just-Eat [5] and independent ‘Fish & Chips’ shops’ websites. The advantages of using food delivery websites as data source is that it is convenient to search ‘Fish & Chips’ shops in each city of the UK by postcode. In addition, each shop which is searched out from food delivery websites is available to crawl data directly that the shop has a valid link and the page of that link has menu content. Whereas when searching on independent websites it may be the case that the desired content (e.g. menu) is only available in a PDF and hence cannot be crawled. Further on independent websites it may be the case that the URLs provide on the site are broken and hence also cannot be crawled. Thus, the dissertation originally planned to use the food delivery website as a data source. However, sites like Just-Eat clearly state that direct crawling of their web site data is not allowed [6]. Therefore, using such a food delivery website to crawl data directly in this dissertation project may be illegal. This fatal flaw meant this method could not be used. As a consequence, Google Maps has been used to find independent websites of ‘Fish & Chips’ shops and the data from these independent websites has then been used as the data source.

Web crawling is the mechanism by which information has been collected from target websites [7]. Specifically, the Python module urllib2 has been used to simulate browser behaviour to download web pages and handle request errors [8] to get the full website HTML data of ‘Fish & Chips’ shops.

## HTML data cleaning techniques

Data cleaning is used for improving the quality of data which is used for subsequent processing through detecting inconsistencies and removing errors [9]. In this project, the dataset required to be cleaned is HTML data. The goal of data cleaning in this project is to obtain independent words (such as ‘haggis’), noun phrases (such as ‘monday supper deal haggis’) and word pairs (such as ‘salad with haggis’ can be divided into ‘salad with’ and ‘with haggis’ word pairs) with shop coordinates from HTML datasets and city dataset which contains coordinates.

In the web-based dataset, there is a lot of content that is not required by this project, such as name, attributes of HTML tags, script code and special symbols. The project only focuses on information which the user can see on the page rather than the implementation details of the page. However, in terms of content which customers can see, there is a lot of redundancy, such as the singular and plural of the same noun all represent the same word. Therefore, the project should not only filter useless content in the HTML data, but also classifying words that represent the same meaning such as ‘chip’ and ‘chips’ into the same category (mainly focuses on the classification of singular and plural nouns with the same meaning). Fortunately, the regular expression, HTMLParser and NLP can help the project to achieve the data cleaning goal.

### HTMLParser

HTMLParser is an open source, fast and robust HTML parsing tool for extracting and cleaning content of HTML [10]. It can customize HTML tag content extraction based on user requirements [11]. In this project, the HTMLParser mainly responsible for data extraction and filtering. The data source used in the project are independent websites that the HTML structure of most websites is different as Fig. 2 and Fig. 3 shows (small parts of the website structure are the same because they are developed by the same company). Fig. 2 shows part of the HTML source code of one of the websites that this website writes all the CSS styles on the page. Fig. 3 shows part of the HTML source code of another website that this page introduces some JavaScript code between the HTML element tags. In addition, these two websites are completely different in the HTML structure of the menu. Therefore, in order to extract content from HTML source code with different structures, the HTMLParser plays an important role. It does not pay attention to the structure of the website design, it only pays attention to the name of the HTML tag, such as ‘div’ and ‘script’. As a consequence, the project can easily filter absolutely useless content based on the tag name, such as the content in the ‘script’ tag and extract potentially valuable content from the remaining tags. However, because the design styles of different web pages are different, the extracted data may also contain special symbols such as field trailing space symbol that will interfere with the cleaning result. Thus, the project also uses the regular expression which is a source language which can locate specific character strings in text [12] to filter the result of the HTMLParser.



**Figure 2: HTML source code with HTML style of one of ‘Fish & Chips’ shops’ websites**



**Figure 3: HTML source code of one of ‘Fish & Chips’ shops’ websites**

### Natural Language Processing (NLP)

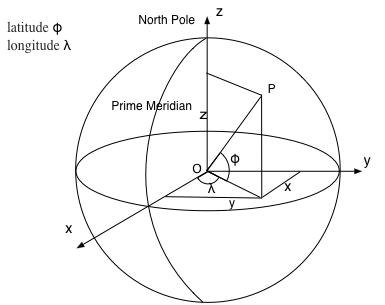
In order to solve the problem of data redundancy in the extracted content, the project uses method of semantic recognition in NLP. Natural Language Processing (NLP) is using computer to understand and manipulate natural text or speech to process tasks [13]. This project mainly wants to change the singular and plural nouns of the same root into singular nouns and the Natural Language Toolkit (nltk) can provide the solution. nltk is an open source tool written by Python with collection of modules and corpora [14]. nltk determines the part of speech of a word based on its corpus and the identification method has been encapsulated which the project can use directly to identify plural nouns and convert them to singular forms. However, in English some words can be both plural nouns and verbs and the nltk will treats all words as nouns and converts them into singular. Fortunately, this project does not care whether the word being converted is a verb or a noun. It only cares whether the word is distributed regionally.

## Geographic data visualisation

One of the aims of the project is to explore geographically distributed features to represent regionality, so the project uses Cartesian coordinate systems for geolocation calculations and Matplotlib for data visualisation.

### Central point calculation algorithm

The core calculation in this project is the set of coordinates’ (the method of obtaining it will be described in detail in 3.1.2) central point which is the centre of all shops which contain a specific content. The importance of the central point is that most regional features found in this project are derived from it. The project uses a set of geographic coordinates containing this content to calculate the centroid. The algorithm used in this project for calculating the central point through combining geographical coordinate system with Cartesian coordinate systems which regards the Earth as a sphere (Fig. 4). This combination is also known as ECEF ("earth-centered, earth-fixed") [15]. In Cartesian coordinates, earth is a sphere centered at the origin [16]. The z axis points to the north pole. The x, y axes are on the equatorial plane that the x-axis passes through the equator and the prime meridian and the y-axis points to the equator at 90 degrees east [17]. However, the coordinates of the central point obtained by using this algorithm in the project are not accurate since the algorithm regards the earth as a sphere rather than ellipse which is the shape of the earth itself. Fortunately, the requirement of the accuracy of the coordinates of the central point in this project is not high, because this project is concerned with the distribution of content.



**Figure 4: geographical coordinate system with a cartesian coordinate systems**

As Fig. 4 shows, point P represents a geographical coordinate with latitude and longitude . A series of coordinates can be represented as latitude , longitude  (i = 1n). Thus, in cartesian coordinate systems, the coordinates of the three directions can be expressed as:

1. =
2. =
3. =

The centroid of these points is the average of the sum of :

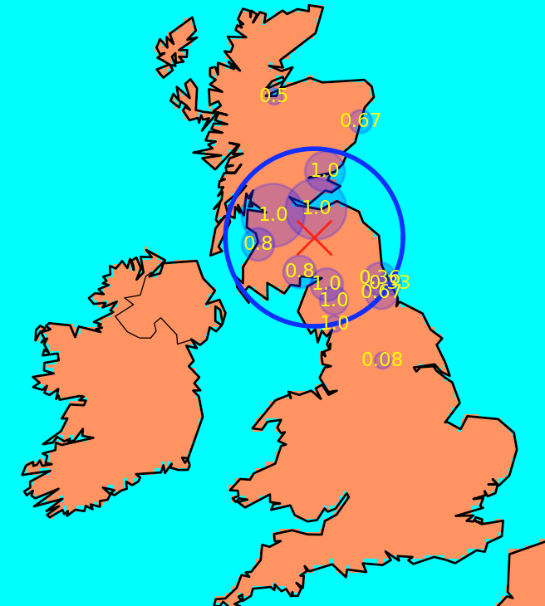
1. () =

The coordinate of the centroid can be expressed as:

1. =
2. =

### Visualisation with Matplotlib

After the project gets the coordinates of central point and other features such as radius (details will be described in 3.1.3), the project hopes to display the distribution of a specific content on the UK map to observe the distribution of the content. The visualisation tool selected by the project is Matplotlib package of Python which is an open source portable Python plotting package used in scientific, engineering and financial fields [18]. It can implement complex data visualisation processes with simple encapsulated methods. Because of the convenience of this tool, most data visualisation processes in the project are achieved by it. Specifically, in this project, Matplotlib package mainly completes the visualising of geographic information distribution and the line graphs. In terms of the visualisation of geographic information distribution, the project uses one of the Matplotlib toolkit named Basemap [19]. Basemap provides a possibility that the project can draw Matplotlib plot over the real-world map [20]. This indicates the Basemap replaces the bottom canvas of the Matplotlib, so it can implement the goal of plotting other graphics such as radius and circumference curve on the map. The following figure (Fig. 5) is an example of using Basemap to visualise the distribution of ‘haggis’ (details will be stated in 3.1.3).

****

**Figure 5: Haggis distribution range**

## Data mining with machine learning methods

This project is an exploratory project, therefore, at the beginning of the project, there is no clear definition of regional features. As a result, the discovery of features and obtaining regional results are the process of data mining. Data mining is a process to extract patterns which represent useful information from massive dataset [21]. In the early stage of the project, the discovery features went through two phases. The first is to find some content widely distributed on the map and some regionally distributed content based on the results of the geographical distribution of content such as Fig. 5. The second phase is to find the commonalities of these widely distributed content and regional content, and the commonalities can be regarded as the features (national or regional) of the content. For example, in the range of 200,000 meters from the central point, the number of shops which have the same regional content will be more than 70% of total shops number which contain that content. However, in terms of most national content, this number will be lower than 60%. As a consequence, in the range of 200,000 meters, the number of shops exceeding 70% of the total number can be regarded as a feature of regional content. After the project uses the above method to get more features, the project tries two machine learning methods to judge the regional content. One of them is the decision tree and other is logistic regression and both of them belong to the method of supervised learning. Supervised learning means that the training data has both features and labels. Through training, the machine can find the connection between the features and the labels by itself and can judge the labels when facing data with only features without labels [22]. In this project, the training data set includes widely distributed content and regional content that are judged in advance based on experiences, and they are all marked on zero (widely distributed) or one (regional).

### Decision tree

A Decision tree is mainly used for classification and prediction of models [23] and the project uses decision tree to classify regional content and widely distributed content. There are two algorithms tested in this project. One is the ID3 algorithm and the other is the Cart algorithm. Both algorithm use training dataset to create the tree and then use the tree to classify the test dataset [24]. The reason why the project chooses these two algorithms is that after the project found some features of regional content, the thresholds of these features’ value were defined by the developer. For example, if the average distance of all shops which contain a specific content more than 300,000 meters from the central point, the content is definitely not regional content. The threshold 300,000 meters is defined by developer and can be regarded as a symbol (‘>300,000’). Thus, the project uses the ID3 algorithm which uses symbolic data to generate the decision tree [25] to classify the content to regional or not. However, there are some disadvantages to use the ID3 algorithm (details in 4.1.2) and the project tries another decision tree algorithm – Cart algorithm (details in 4.3.1).

In this project, the result of the decision tree is binary (details in 4.2 and 4.3), which means content is judged to be regional or non-regional. However, the project also wants to know how much probability content is judged as regional. Thus, the project tries another method - logistic regression.

### Logistic regression

The project studies the classification problem, so the dependent variable of the model is classification variable (0 or 1) and the independent and dependent variables of the model are nonlinear. As a result, logistic regression model is more suitable for this project and the project selected the logistic regression model of the Sklearn-learn package as the classifier to find the possibility that each piece of content is judged to be regional.

Logistic regression is well suited to describe the relationship which is expressed as probability between classification results and one or more classifications [26]. It can adapt to multiple classification results. In this project, logistic regression is used to calculate the probability of a binary event occurring under multiple independent features [27]. The following model is the model of logistics regression:

denotes the vector of feature variables, and denotes the associated binary output. represents the weight vector and the is the transposed matrix of w. is the sigmoid function. is the intercept. The logistic regression has model:

Logistic loss function ( is ):

Supposing the training dataset is and the average logistic loss:

Logistic regression problem:

Overfitting problem: in supervised learning when there are many input features, but only a small number of key features determine the classification target. That is, when the number of training set data is insufficient, the classification model may perform well on the training dataset but not well on the test dataset [28]. Thus, when there are many features, overfitting will become a problem of the model unless the training set is ample [29]. In order to solve this problem, L1 and L2 regularizations were used.

L1 regularization [28]:

Lasso (L1) penalty encourages the sum of the absolute values of the to be small [29]. It uses sparsity to fit model with many features [30]. The sparsity means that L1 penalty will automatically filter some features that have less impact on classification. L1 penalty achieves the filtering by reducing the regression coefficient to 0 and slightly reducing other regression coefficients [31].

L2 regularization [28]:

L2 penalty encourages the sum of the squares of the to be small [29]. It will reduce the regression coefficient but will not be zero [31]. Thus, L2 penalty will weaken the dominant classification feature and enhance the influence of other features. If each feature has an effect on the classification, L2 penalty is more suitable.

In this project, using logistic regression and its L1 and L2 regularizations can not only help the project to obtain probabilities but also help the project to see the impact of each feature on the classification results.

# Iteration 1

In iteration one, the project will do the following tasks: data acquisition, data cleaning and data visualisation. The data acquisition procedure is mainly responsible for crawling and storing menu websites. The data cleaning process is primarily responsible for obtaining independent single words from crawled HTML files. The data visualisation phrase is divided into two parts. One of them is geographical map visualisation and another is trend visualisation (details in 3.1.3). The aim of the iteration one is to find regional features of regional words through observing the result of some known regional words’ geographical distribution map and trend graph. In addition, iteration one details the process of discovering these features and makes plans for the next iteration.

## Methodology

Iteration one is mainly focuses on four aspects, which are web data acquisition, HTML data cleaning, geographic data and ratio trend visualisation and exploration of features.

### Data acquisition

After the project identified the data source is independent ‘Fish & Chips’ shops’ websites, the project started to find URLs of these websites through searching on the Google Maps and other food recommendation websites. The method for searching websites is first finding the city, then searching for ‘Fish & Chips’ and get websites URLs from the searching result. In the beginning of the project, the project collected websites of ‘Fish & Chips’ shops in some densely populated cities in the UK, such as London, Manchester and Glasgow. However, the project found that these collected shops are concentrated in the north-central (e.g. Edinburgh, Glasgow) and south-central (e.g. Manchester, Sheffield), with few shops in the north (e.g. Dundee, Inverness), southwest (e.g. Plymouth) and central regions (e.g. Newcastle). In order to solve the problem of uneven distribution of shops, the project added some shops which distributed in northern, central, and southwestern cities from the Google Maps.

The initial goal of the project is to obtain a collection of shops that their distribution can cover all parts of the UK. However, in the UK, although there are a lot of ‘Fish & Chips’ shops, not every ‘Fish & Chips’ shop offers the menu website. Besides, as 2.1 described, some of these ‘Fish & Chips’ websites cannot be crawled. As a consequence, the project finally collected two hundred and forty available websites of the ‘Fish & Chips’ shop. The distribution of the shops which contain these websites basically covers most parts of the UK. However, although the distribution of shops covers most cities, the number of shops in each city still shows bias. That means most of the shops are concentrated in densely populated cities, and other cities with sparse populations such as Inverness and Carlisle have fewer shops. Thus, this imbalance will be reflected when the project visualises the shops' geographic data in 3.1.3. Each ‘Fish & Chips’ shop collected by the project will be assigned Id, city name, and the URL and this information is stored in an CSV format file. Id is used to uniquely identify the shop and the city name is used to find the coordinates of the city where the shop is located (cities with their coordinates are stored in another file created by the project).

After the project completes the shop collection, the project wrote a Python script that uses urllib2 module to crawl HTML data from the collected websites. The urllib2 module provides a way to simulate a browser to send HTTP requests to a website. This method avoids the problem of some websites' denying access due to the detection of abnormal access. Besides, the script uses file which has the coordinates of different cities and finds the geographic coordinates of each shops. (In this project, the coordinates of the shops in the same city are the coordinates of the city).

The script generates multiple independent files, each of which stores the HTML source code of a shop, and the file name is the Id of the shop. In addition, the script also generates a mapping file which contain id, name of HTML source code file and coordinates of that shop. As a result, after the data acquisition procedure, the project connected each shops’ HTML content with its coordinates.

### Data cleaning

The entire process of data cleaning in this project is a process similar to Map-Reduce that the Map method generates a series of intermediate key and value pairs and the Reduce method merges the results of the Map method according to the same key value [32]. The reason why data cleaning procedure is similar to the Map-Reduce is that this project extracts the content from the HTML source code in the form of key (content)-value (shop id and a set of coordinates) and merge the content with the same key. This project uses two Python scripts to achieve the Map and Reduce processes to complete the data cleaning procedure. The Map script is responsible for extracting independent single words with their shop id (its role will be described in the Reduce script) and shop coordinates. The Reduce script is mainly responsible for merging the words output by the Map script and adding the coordinates of the same word to the coordinate set of that word.

The Map script mainly focuses on filtering useless HTML content, extracting the content in the HTML tag, splitting the content into separate independent words and converting all plural nouns to singular. Firstly, the script reads the mapping file generated in 3.1.1 and sequentially reads the HTML file that the project crawls and hand HTML content to the HTMLParser for processing. The HTMLParser firstly filters the tags with the content in them that project do not need. The useless tags names are defined by the project and in this project, useless tags are ‘script’, ‘style’, ‘link’, ‘head’, ‘a’ and ‘title’, since most of the content in these tags is HTML code. Next, the HTMLParser will sequentially recognize the names of other tags with their content, but this project only focuses on the content. In order to extract independent single words, the project uses regular expressions and *split()* function in the function which HTMLParser handles the contents of the HTML tag. The project will first use regular expressions to filter numbers and process special symbols such as ‘.’, ‘+’ and ‘-’, and then use the segmentation method to split the content into separate independent words and convert them to lowercase.

The project originally wanted to convert plural nouns into singular nouns after getting the independent single words. However, the ASCII encoded special spaces appear in some of independent single words and the regular expressions cannot recognize them. The reason for these special spaces is the inconsistency between the character set encoding of some websites and the compiler character set encoding. As a consequence, if the project directly converts the part of speech, plural nouns with special spaces will be treated as proprietary singular nouns. Further, the project finds that these words with special spaces will only display the ASCII code of the special space such as ‘\xc2\xa0’ after added to the list. Thus, the project first saves all the results of HTMLParser to the output list. Each row of the output list includes shop Id, independent single word (may contain special spaces) and the shop coordinates. Next, the project will convert output array to string, and the ASCII code of special spaces will be existed as strings, and then the project can use the regular expression to filter special spaces. After filtering the special spaces, the project reconverts the string of the output list to a list and used as the output of the Map script.

Before the Reduce script processing the output data of the Map script, the project uses the Unix *sort()* method to sort the independent single word column of the output list of the Map script. This will reduce the workload of the Reduce script that when processing each row, the Reduce script does not have to determine whether the word in current row appear in the words that have appeared before. The script only needs to determine whether the word in the current row is the same as the word in the previous row. If the same, add the coordinates of the current word to the previous coordinate set. If they are different, a new coordinate set is created for the current word, and the current coordinate is added to the new set. Besides, in this project, each word is only allowed to appear once in one shop. That means in terms of same words with the same shop id which is one of the output value of Map script, only one set of coordinates can be added to the word coordinates collection. The reason for this is because the coordinates of the word added to the collection can represent the coordinates of all the same words in a shop.

The output of the Reduce script is a CSV format file which each row is an independent single word with its coordinates collection. As a result, this file can be used for calculating the central point of each word’s distribution and the data cleaning procedure has finished.

### Data visualisation

In iteration one, the data visualisation of this project is divided into two parts. One of them is geographic maps data visualisation, and another is trend of the number of shops which contain a specific word increasing with distance (meter) from the centre point. The project wants to find the features of regional words through the results of these data visualisations.

In terms of the geographic maps, the project considers this to be the most intuitive way to see if a word is a regional word. Before visualising the geographic maps, firstly, the project will calculate the central point of a word distribution based on the coordinates set results of the word. Secondly, the project will calculate the radius of the word distribution. The method of calculating the radius is calculating the distance between all word’s coordinates from the centre point and find the largest distance as the radius. The distance between the central point and each coordinate point is derived from the Euclidean distance, because the Basemap has converted the Earth's sphere into a plane. Next, the project calculates the number of shops in the city that contain the word and the proportion of those shops in the total number of shops in the city. By this way, the geographic map can not only show the distribution of the word but also show the uneven distribution of the number of shops in each city. This can help users better understand the details of word distribution. However, there is a problem that some outliers which means few shops are far from the central point will have huge impact on the above parameters, especially on the radius that the radius will become very large because it covers all the shops. In order to filter the outlier shops, the project will firstly calculate the central point of shops which contain the word and then sorts the distances of all shops. Next, the project sets a percentage that the project will only take a percentage of the shops close to the central point and then the project will recalculate the central points and other parameters based on these selected shops. The Fig. 5 is an example of the distribution of ‘haggis’ which contain 95% shops (the project will explain why setting 95% as the percentage in 3.2). In Fig. 5, ‘X’ represents the central point of the distribution of ‘haggis’. The bold blue line represents the circumference of the distribution. The size of many blue small circles in the figure represents the number of shops which contain ‘haggis’ in each city. The decimal in each blue circle means the number of shops in the city that contain the word as a percentage of the total number of shops in the city.

The reason why the project wants to visualise the trend is that trend graph is a conjecture of the project for the distribution trend of words with regional features. If a word is a regional word, as the distance from the central point becomes larger, the number of shops will increase to a certain value and then no longer grow. Besides, when the distance from the central point begins to increase, the number of shops of regional words will increase rapidly, and as the distance increases to a certain distance, the growth of the number of shops will slow down. The project believes that the ratio of the number of shops within this particular distance to the total number of shops can be considered as a feature of regional word. As a result, the project compared some known regional and non-regional words and found ‘ratio’ features from them. ‘ratio’ means the number of shops whose distance is less than specific meters from the central point/ total shop number.

## Findings

The following diagrams show geographic map visualisation and the trend visualisation results of regional words and national words. In terms of geographic map visualisation, the reason for selecting 95% shops of a word to calculate parameters is that after adjusting the percentage, the project found that selecting 95% can filter out almost all outliers that have huge impact on the results. In addition, this percentage can retain all normal distribution shops.

In iteration one, the project obtained the features of some regional word by comparing the results of pre-known regional words with the results of pre-known national words.

### Regional words findings

|  |  |
| --- | --- |
|  |  |
| **Figure 6: ‘haggis’ distribution (95%)** | **Figure 7: The number of ‘haggis’ shops varies with distance** |
|  |  |
| **Figure 8: ‘bru’ distribution (95%)** | **Figure 9: The number of ‘bru’ shops varies with distance** |
|  |  |
| **Figure 10: ‘naan’ distribution (95%)** | **Figure 11: The number of ‘naan’ shops varies with distance** |
|  |  |
| **Figure 12: ‘roe’ distribution (95%)** | **Figure 13: The number of ‘roe’ shops varies with distance** |
|  |  |
| **Figure 14: ‘supper’ distribution (95%)** | **Figure 15: The number of ‘supper’ shops varies with distance** |
|  |  |
| **Figure 16: ‘pakora’ distribution (95%)** | **Figure 17: The number of ‘pokora’ shops varies with distance** |

According to the regional words findings, the project found that the number of shops increased greatly within 200,000 meters from the central point. After 200,000 meters, the growth trend of the number of shops shows a slowdown. This confirms the hypothesis of the project that regional words show a rapid growth trend within a certain distance and in this project, the certain distance is 200,000 meters. Besides, the project found that the ‘ratio’ of known regional words is greater than 65%. Thus, the project wants to use ‘ratio > 65%’ as a feature of regional words. However, there is a problem that most of words with few shops (less than ten shops) are distributed within 200,000 meters. Besides, the ‘ratio’ of these words is 1. For example, Fig. 18 and Fig. 19 show the distribution and trend of ‘massala’ which just has four shops. In Fig. 18, ‘massala’ looks like a regional word and according to Fig. 19, ‘massala’ has the feature of regional word. However, the distribution sample of ‘massala’ is really too small that the project cannot directly determine that 'massala' is a regional word. As a consequence, the project decided that for words containing only ten or less stores, the projects treat them as non-regional words.

|  |  |
| --- | --- |
|  |  |
| **Figure 18: ‘massala’ distribution (95%)** | **Figure 19: The number of ‘massala’ shops varies with distance** |

### National words findings

|  |  |
| --- | --- |
|  |  |
| **Figure 20: ‘chip’ distribution (95%)** | **Figure 21: The number of ‘chip’ shops varies with distance** |
|  |  |
| **Figure 22: ‘sausage’ distribution (95%)** | **Figure 23: The number of ‘sausage’ shops varies with distance** |
|  |  |
| **Figure 24: ‘supreme’ distribution (95%)** | **Figure 25: The number of ‘supreme’ shops varies with distance** |
|  |  |
| **Figure 26: ‘gift’ distribution (95%)** | **Figure 27: The number of ‘gift’ shops varies with distance** |
|  |  |
| **Figure 28: ‘soup’ distribution (95%)** | **Figure 29: The number of ‘soup’ shops varies with distance** |
|  |  |
| **Figure 30: ‘daily’ distribution (95%)** | **Figure 31: The number of ‘daily’ shops varies with distance** |

The above figure shows three types of non-regional words. Fig. 20 – Fig. 23 show the distribution and trend of national words which distributed in almost every shop in every city. Fig. 24 – Fig. 27 show words which distributed in almost every city, but the number of shops contained that word in each city is not a lot. Fig. 28 - Fig. 31 show words which not distributes in many cities, but widely distributed throughout the UK. According to these national words findings, the project found that the number of shops will not increase obviously when the distance start to grow. This is more certain that ‘ratio’ is a feature of regional words rather than non-regional words.

## Evaluation and Improvement

Through the above findings, it can be explained that ‘ratio’ can be regarded as one of features of regional word, but it is not enough to only rely on ‘ratio’ to judge all regional words such as ‘funghi’. According to Fig. 32 and Fig. 33 which are the distribution and trend of ‘funghi’, the project found that the number of shops which contain ‘funghi’ rises smoothly and the ‘ratio’ of ‘funghi’ is just 50%. However, as Fig. 32 shows, there are many shops that contain this word in a small area and it looks like a regional word. Thus, if ‘funghi’ is only judged by ‘ratio’, it must be treated as a non-regional word. As a consequence, the project requires to discover more features to make more accurate decision.

|  |  |
| --- | --- |
|  |  |
| **Figure 32: ‘funghi’ distribution (95%)** | **Figure 33: The number of ‘funghi’ shops varies with distance** |

According to the results of ‘funghi’, the project first thought of calculating the average distance of the shop which contain the word from the central point. This is because the small average distance from the central point means that the word distribution range will not be very large. Besides, according to the known regional words results, the average distance from the central point of regional words is below 300,000 meters. Thus, the project decides to use ‘average distance < 300,000’ as a feature of regional words.

In the exploring regional features process of the project, the project is inspired by the phenomenon that the closer to the central point, the denser the distribution of the shops. Thus, the project decided to use the median distance of all shop which contain the word to try to discover new features. Fortunately, the project found that all known regional words have a ‘proportion’ feature which means the number of shop whose distance from the central point less than the median distance/ the total number of shops > 67%. In addition, the project also found that the number of cities with a regional word is no more than 19. Thus, the project regards the ‘city number’ < 19 as a feature of regional words.

At the end of iteration one, the project calculated the values of the above five features for all words and stores them in CSV format (word, ratio, proportion, average distance, city number, the number of shops).

## Summary and Future Work

Through observing and comparing geographical maps and trends of known regional and non-regional words, iteration one derives five features (‘ratio’ > 65%, ‘shop number’ > 10, ‘average distance’ < 300,000 meters, ‘proportion’ > 67%, ‘city number’ < 19) of regional words. The project wants to use these features to make regional judgement for all the separated independent words. However, the project cannot use all these features in one conditional statement to judge regional words, because some words such as ‘massala’ only satisfy some of these features. As a result, the project can only divide a data set into two parts by selecting one feature at a time, and then divide the result of the division through another feature. However, the project cannot judge whether each division is the optimal division which means the currently selected feature can maximize the distinction between regional and non-regional words. Fortunately, ID3 algorithm in decision tree can help the project to solve this problem that ID3 algorithm uses Entropy to select feature to achieve optimal division. Thus, in next iteration, the project will use decision tree to use these features to get the regional words result.

# Iteration 2

The aim of the iteration two is to use the decision tree and features found in iteration one to classify the independent single words. In this iteration, the project will do the following tasks: generating the training set for the decision tree, using ID3 algorithm to generate the decision tree, evaluating the ID3 algorithm and the regional result of the ID3 decision tree. In addition, after evaluating the ID3 algorithm, this iteration introduces another decision tree algorithm (Cart algorithm) and evaluates the algorithm and regional result and makes recommendations for the next iteration.

## Methodology

In this section, the project focuses on the generation process of the training data set and the ID3 algorithm.

### Training dataset

The training dataset is generated by the project, containing some known regional words and national words and the project initially wanted to use 20% of the total data (5289 words) as the training dataset. According to the ‘word with features’ file generated from the iteration one, the project found that only one-fifth of the total words’ shops number is more than ten and among these words, most words show obviously non-regional features. Thus, finding a sufficient number of non-regional words as training data is easy. However, it’s difficult to find a sufficient number of regional words by observing the data set in the ‘word with features’ file. That because finding regional words not only requires to observe the geographic maps visualisation result but also influenced by the cognitive of the developer. In the process of generating a training set for the first time, the cognition of the developer becomes the biggest obstacle to select regional word, because there are many seemingly common words such as ‘securely’ and ‘instantly’ have some of regional features. The project though that although these words have some regional words features, they do not have all the features. As a consequence, the project did not add these words with their features into the training dataset. In the end, the project just defined twenty-five regional words, most of them were dish names and others were place names such as ‘yorkshire’. Besides, there were seventy non-regional words were added to the training dataset, then the project used this dataset to generate the ID3 algorithm decision tree.

### ID3 Algorithm

D3 algorithm constructs decision tree by selecting most useful features [33]. These features can make the classification of data set more effective. Thus, the project requires an algorithm to measure the suitability of features and select features. The Entropy can measure the impurity of training dataset [34] that the greater the entropy, the more complex the information. As a consequence, the project can use the information gain which is the amount of entropy lost by adding a feature to select representative features.

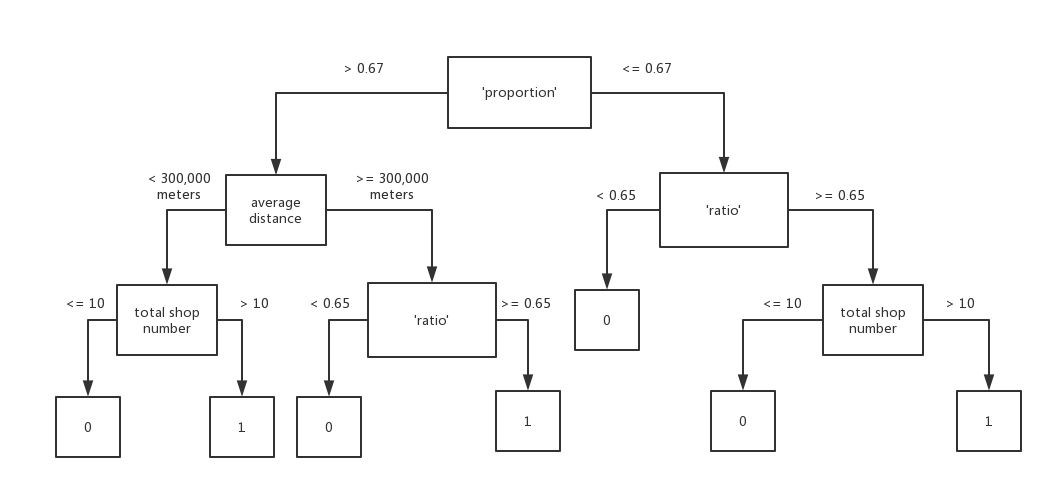
Entropy:

Information Gain: a represents a feature.

The decision tree construction process of ID3 algorithm is divided into the following steps:

* Loading training dataset generated in the 4.1.1.
* Calculating the Entropy.
* Data segmentation based on optimal segmentation feature.
* Selecting the best segmentation feature based on the maximum information gain.
* Recursively building a decision tree.
* Sample classification.

## Findings

****

**Figure 34: ID3 algorithm decision tree**

Fig. 34 is the decision tree result of ID3 algorithm, where ‘0’ represents non-regional words and ‘1’ represents regional words. There are 55 following independent words are judged as regional words:

‘haggis’, ‘irn’, ‘bru’, ‘kiev’, ‘inferno’, ‘crunch’, ‘skate’, ‘bolognese’, ‘macaroni’, ‘naan’, ‘hamburger’, ‘plaice’, ‘rib’, ‘kidney’, ‘spaghetti’, ‘carbonara’, ‘pasty’, ‘roe’, ‘balty’, ‘meaty’, ‘cucumber’, ‘cob’, ‘guava’, ‘pakora’, ‘pukka’, ‘savoury’, ‘pattie’, ‘burdock’, ‘parmesan’, ‘splash’, ‘dandelion’, ‘scallop’, ‘keema’, ‘samosa’, ‘sury’, ‘rump’, ‘dazs’, ‘give’, ‘smokey’, ‘macaroni’, ‘cornish’, ‘bit’, ‘quattro’, ‘passion’, ‘facebook’, ‘chosen’, ‘value’, ‘securely’, ‘yorkshire’, ‘instantly’, ‘rock’, ‘shot’, ‘haagen’, ‘bull’.

## Evaluation and Improvement

Although the project got regional word, in addition to the dish words, many of these words are seemingly common such as ‘bit’, ‘give’ and ‘value’. In order to understand why these words were judged as regional words, the project decided to find the context of these words to figure out the usage of these words in the website. To achieve this, the project wrote a Python script whose input are these regional words. This script is responsible for searching the context of these regional words in all the HTML files obtained in the iteration one and generates a file of these words and their context. Thus, the project can according to this file to find the reason why the words were judged as regional words.

According to the word context file, the project found that the following words represent a kind of dish or part of the name of the dish: ‘haggis’, ‘irn’, ‘bru’, ‘kiev’, ‘inferno’, ‘crunch’, ‘skate’, ‘bolognese’, ‘macaroni’, ‘naan’, ‘hamburger’, ‘plaice’, ‘rib’, ‘kidney’, ‘spaghetti’, ‘carbonara’, ‘pasty’, ‘roe’, ‘balty’, ‘meaty’, ‘cucumber’, ‘cob’, ‘guava’, ‘pakora’, ‘pukka’, ‘savoury’, ‘pattie’, ‘burdock’, ‘parmesan’, ‘splash’, ‘dandelion’, ‘scallop’, ‘keema’, ‘samosa’, ‘sury’, ‘rump’, ‘dazs’, ‘rock’, ‘shot’, ‘haagen’, ‘bull’, ‘macaroni’, ‘cornish’, ‘quattro’, ‘passion’. It is worth mentioning that in these regional words, ‘irn-bru’ always uses together which represents a Scottish drink; ‘haagen’ and ‘dazs’ represent regional words because in that area, Haagen-Dazs maybe has more trade links with the merchants; ‘rock’ always used with ‘eel’. ‘rock eel’ represents a kind of fish; ‘shot’ always used with ‘hot’ and ‘hot shot’ represents a kind of dishes; ‘bull’ always used with ‘red’ and ‘red bull’ is a drink and the reason why ‘bull’ is regionally distributed maybe same as ‘haagen’; ‘cornish’ is always used with ‘pasty’ and ‘cornish pasty’ is a dish; ‘macaroni’ is always used with ‘cheese’ and ‘macaroni cheese’ is a dish; ‘quattro’ is always used with ‘stagioni’ and ‘quattro stagioni’ is a kind of pizza; ‘passion’ represents fruit or dish. Maybe in that area this kind of fruit is famous or selling well.

In addition to the regional dish words mentioned above, there are some words people always used in daily life are judged as regional words and the project has found the reason for this based on these words’ context. ‘securely’ is mostly used with ‘with’ and ‘pay securely online’. Besides, the project found that when ‘securely’ used with ‘with’, all websites that use this usage have the same style. Similarly, all websites which have the usage of ‘pay securely online’ have the same style. This may be because the website of shops in the area was developed by the same company. As a consequence, ‘securely’ appears regionally. Similarly, the reason why ‘chosen’ (always used with ‘flavour’) and ‘instantly’ (always used with ‘chip shop takeaway - order online instantly’) were judged as regional words is same as ‘securely’. In terms of ‘yorkshire’, most of ‘yorkshire’ represent a place named ‘yorkshire’. Therefore, ‘yorkshire’ is a regional word that represents a place name. ‘give’ is used as a verb. Maybe in that area, people are used to expressing their own dishes in this way, such as give the best taste. ‘smokey’ is used as an adjective, usually in conjunction with a ‘bbq’ or ‘sausage’. ‘bit’ is usually used as a degree adverb. ‘value’ is always used with ‘box’ or ‘meal’ which represent dishes. In terms of ‘facebook’, the project found that some websites provide a Facebook account and shops which contain ‘facebook’ are distributed in a small region.

According to the results of the ID3 algorithm, although the project successfully determined the words’ regionality according to the regional features, the project also found a problem. This problem is that the threshold of each feature may not accurate, because these thresholds are defined by the developer by observing a limited amount of data in iteration one. As a consequence, the project wants to use an algorithm to define the feature threshold automatically. However, the project cannot use ID3 algorithm to find the feature threshold, because the limitation of the ID3 algorithm is that it can only deal with discrete values [35]. That means the feature values must be classified based on numerical variables and the project has to mark each training data’s feature to numerical variables. For example, the ‘average distance’ of ‘haggis’ is 216926 meters and the project will mark it as ‘<300,000’. This also affects the efficiency of code execution because the project has to mark symbol for each feature value for each word. Fortunately, Python provides a toolkit (Sklearn-learn) which integrates a variety of machine learning algorithms for supervisory and unsupervised problems [36]. This toolkit can help the project to achieve the goal of finding thresholds automatically. As a result, the project used the Sklearn-learn package decision tree algorithm which uses a kind of optimised Cart algorithm [37] to generate the decision tree, including classification tree and regression tree. In this project, the classification tree is more suitable, because the target of the decision tree is binary.

### Cart Algorithm

Cart algorithm uses binary recursive partitioning procedure to split datasets [38]. In classification tree, Cart algorithm uses Gini index as a property to determine partitioning [39]. The Gini index indicates the uncertainty of the sample. The larger the Gini index, the greater the uncertainty of the sample set which means the probability of the sample belongs to a class is low. In terms of each feature, the Cart algorithm will traverse all possible splitting methods and select the feature which has minimum Gini index as the division criteria [40]. The following formulas shows the calculating of the Gini index.

Assuming that there is a K class, the probability that the sample point belongs to the K class is , then the Gini index is defined as:

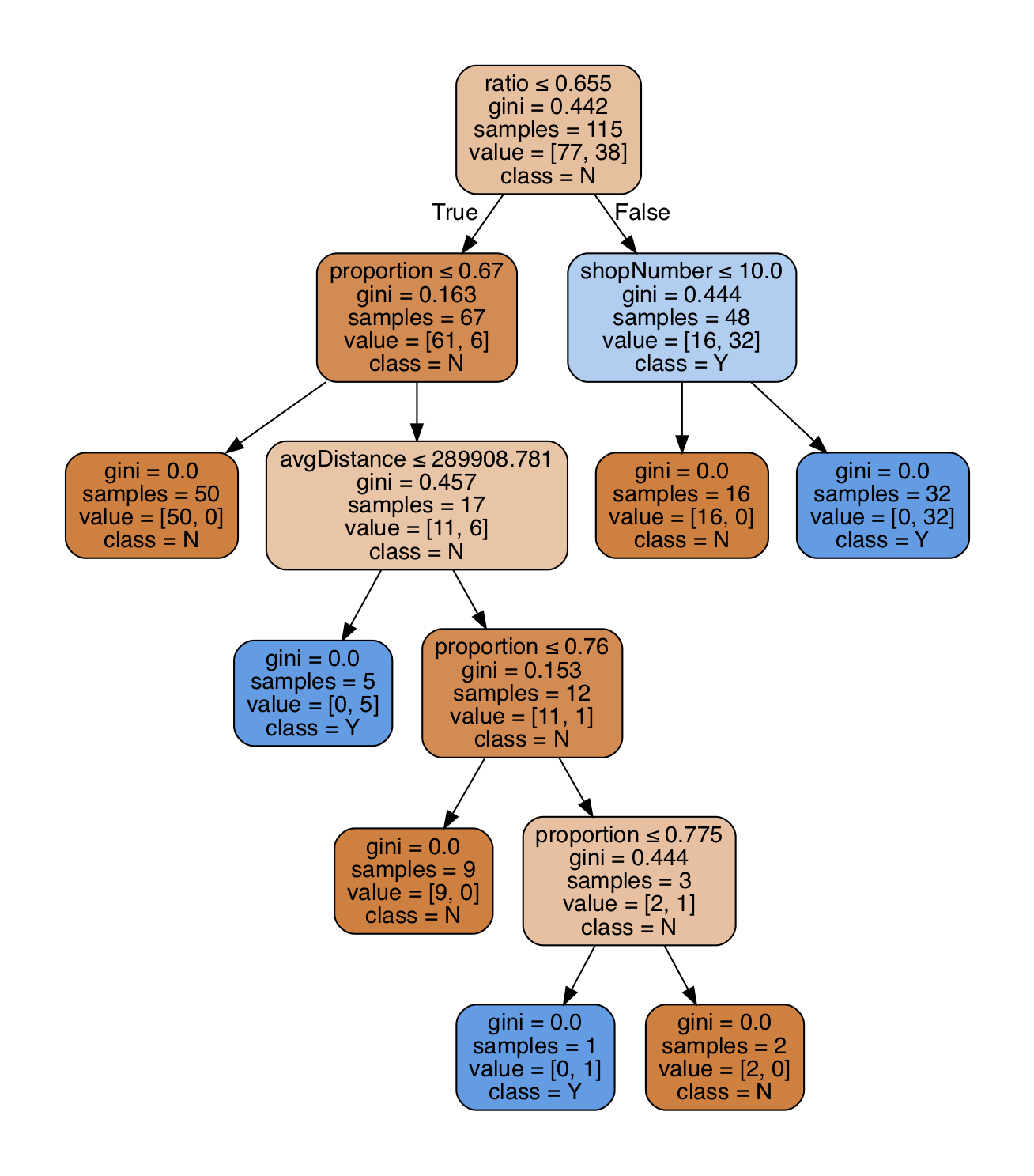
Assuming that be the subset of samples belonging to the k class in D, then the Gini index is:

Assuming that feature A divide the sample D into two data subsets D1 and D2, then the Gini index of the sample D under the feature A is:

The steps to generate a decision tree using the Cart algorithm are as follows:

* Using each feature A in the sample D and each possible value of A (A>=a and A<a) to divide the sample into two parts and calculate the Gini (D, A).
* Find the optimal segmentation feature which has the minimum Gini (D, A). Next, determining whether the splitting stop condition is satisfied. If not, output the optimal segmentation point.
* Recursive call (1) (2)

### Cart Algorithm Findings and Evaluations



**Figure 35: Cart algorithm decision tree**

The Fig. 35 is the decision tree result of using Cart algorithm. Before generating the Cart algorithm decision tree, the project added some regional words to the training dataset based on the results of the ID3 algorithm. The purpose of doing this is to get a better tree. The purpose of this is to get a tree that can classify regional words more accurately.

In terms of the classification result of the Cart algorithm, Although the Cart algorithm generates more tree branches and the branching conditions become more precise, the classification results are not quite different from the ID3 algorithm. There are 56 words were judged as regional words that in these 56 words, excepting ‘mince’, 55 are in the result of ID3 algorithm. ‘mince’ is a word which widely distributed near Edinburgh. However, when using ID3 algorithm, because of its ‘proportion’ larger than 67%, but its ‘ratio’ lower than 65%, so it was judged as a national word.

## Summary and Future Work

In this iteration, the project first obtained regional words. Besides, the project got the reason why the word is judged as a regional word based on the context in which the word appears. Further, the project found that ‘city number’ did not have any impact in the decision tree, therefore ‘city number’ is a useless feature for the project. However, the project wants to continue to explore the possibility of each classification result such as how likely is ‘haggis’ to be classified as regional content. Unfortunately, the project cannot get the probabilities through using the decision tree, because in this project, the result of the decision tree is binary. As a consequence, in the next iteration, the project wants to use a regression classifier of Sklearn-learn package to get the probability that a particular content is in a category.

## Some results

Here are some results.

### More results

When showing results you are likely to use tables and graphs. You can create tables easily in *Word*. To insert a caption for a table such as the one in Table 1, place your cursor in the table, and then choose *Insert > Reference > Caption*. From the *Label* menu in the dialog box that appears, choose *Table*.

|  |  |  |
| --- | --- | --- |
| **File names** | **Satellite** | **Resolution** |
| worldr | Meteosat | 5km |
| worldg | Meteosat | 5km |
| worldb | Meteosat | 5km |

Table 1: This is a simple table. More complicated tables can have headings which pass over more than one column.

From many MS Windows programs you can simply cut and paste images into your document. You can select a caption in a similar way to the caption in the table above (but choose *Figure* instead of *Table*).

One way to produce simple graphs is to use gnuplot which can produce output in various formats. You can output in png format and use *Insert > Picture > From File…* as described above. Figure 2 was produced using gnuplot. For more details, see gnuplot’s internal help system (try, help display and help output).

Another approach is to draw simple ﬁgures using *xﬁg* which allows you to export your results in many formats including *jpeg* and *png*.

*Word* can also import graphics in *eps* format.

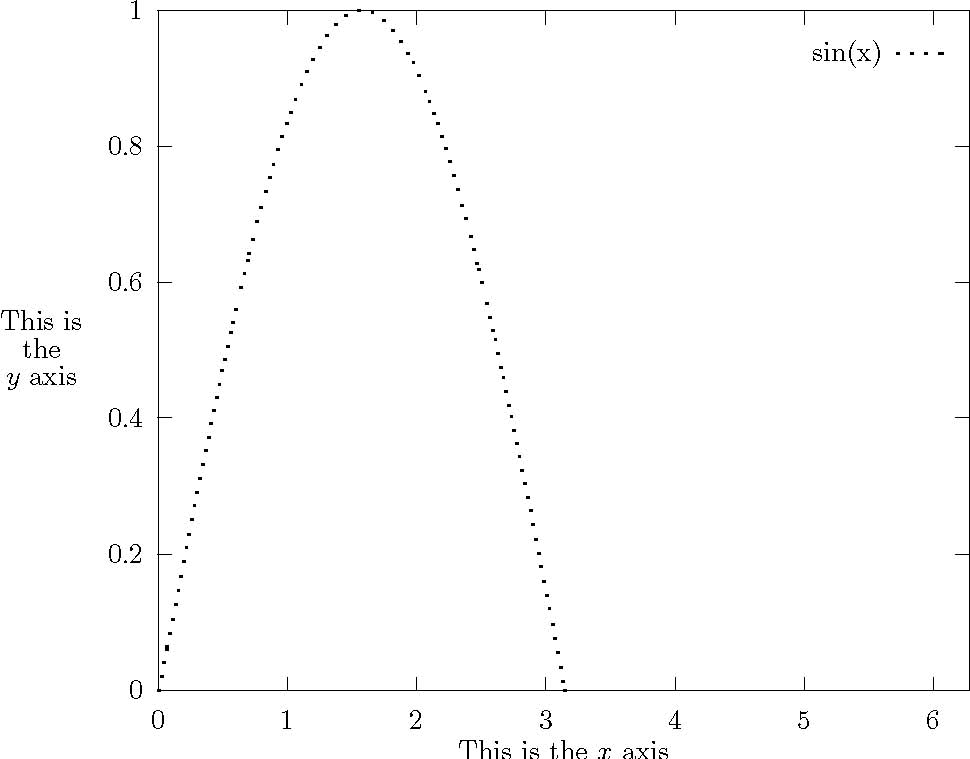


Figure 7: Simple Gnuplot example

# Conclusions

This is the place to put your conclusions about your work. You can split it into different subsections if appropriate. You may want to include a subsection of future work which could be carried out to continue your research.

1. Stuff which is too detailed

Appendices should contain all the material which is considered too detailed to be included in the main body but which is, nevertheless, important enough to be included in the thesis.

1. Stuff which no-one will read

Some people include in their thesis a lot of detail, particularly computer code, which no-one will ever read. You should be careful that anything like this you include should contain some element of uniqueness which justifies its inclusion.

References

Vasumita S Adarsh. (2013, December 26). TastyKhana launches Google map feature for website.(Internet). *The Economic Times*, p. The Economic Times, Dec 26, 2013.

O'Connor, P. (2010). Managing a hotel's image on TripAdvisor. *Journal of Hospitality Marketing & Management*, *19*(7), 754-772.

Hand, D. J. (2007). Principles of data mining. *Drug safety*, *30*(7), 621-622.

Fish and chips. (2010). *Nutrition & Food Science,* *40*(6), 157-165.

*Just Eat*. Online at <https://www.just-eat.co.uk/> (referenced 19/08/2018).

*JUST EAT WEBSITE TERMS AND CONDITIONS.* Online at <https://www.just-eat.co.uk/termsandconditions> (referenced 19/08/2018).

Castillo, C. (2005, June). Effective web crawling. In *Acm sigir forum* (Vol. 39, No. 1, pp. 55-56). Acm.

Goerzen, J. (2004). Web Client Access. In *Foundations of Python Network Programming* (pp. 113-126). Apress, Berkeley, CA.

Rahm, E., & Do, H. H. (2000). Data cleaning: Problems and current

LI, W., & HUANG, Y. (2007). Web information extraction based on HtmlPaser [J]. *Ordnance Industry Automation*, *7*, 024.

Lin, S., & Hu, Y. (2010, July). An approach of extracting web information based on htmlparser. In *Information Technology and Computer Science (ITCS), 2010 Second International Conference on* (pp. 284-287). IEEE.

Thompson, K. (1968). Programming techniques: Regular expression search algorithm. *Communications of the ACM*, *11*(6), 419-422.

Chowdhury, G. G. (2003). Natural language processing. *Annual review of information science and technology*, *37*(1), 51-89.

Bird, S., & Loper, E. (2004, July). NLTK: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions* (p. 31). Association for Computational Linguistics.

Zhu, J. (1994). Conversion of Earth-centered Earth-fixed coordinates to geodetic coordinates. *IEEE Transactions on Aerospace and Electronic Systems*, *30*(3), 957-961.

Clynch, J. R. (2006). Earth coordinates. *Electronic Documentation, February*.

Montenbruck, O., Gill, E., & Terzibaschian, T. (2000). Note on the BIRD ACS Reference Frames.

Barrett, P., Hunter, J., Miller, J. T., Hsu, J. C., & Greenfield, P. (2005, December). matplotlib--A Portable Python Plotting Package. In *Astronomical data analysis software and systems XIV* (Vol. 347, p. 91).

Whitaker, J. (2011). The Matplotlib Basemap Toolkit User’s Guide. *Matplotlib Basemap Toolkit documentation, February*.

Tosi, S. (2009). *Matplotlib for Python developers*. Packt Publishing Ltd.

Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.

Guo, & Koelsch. (2015). The effects of supervised learning on event-related potential correlates of music-syntactic processing. *Brain Research,1626*, 232-246.

Jin, C., De-Lin, L., & Fen-Xiang, M. (2009, July). An improved ID3 decision tree algorithm. In *Computer Science & Education, 2009. ICCSE'09. 4th International Conference on*(pp. 127-130). IEEE.

Hssina, B., Merbouha, A., Ezzikouri, H., & Erritali, M. (2014). A comparative study of decision tree ID3 and C4. 5. *International Journal of Advanced Computer Science and Applications*, *4*(2).

Umanol, M., Okamoto, H., Hatono, I., Tamura, H. I. R. O. Y. U. K. I., Kawachi, F., Umedzu, S., & Kinoshita, J. (1994, June). Fuzzy decision trees by fuzzy ID3 algorithm and its application to diagnosis systems. In *Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on* (pp. 2113-2118). IEEE.

Peng, C. Y. J., Lee, K. L., & Ingersoll, G. M. (2002). An introduction to logistic regression analysis and reporting. *The journal of educational research*, *96*(1), 3-14.

Walker, S. H., & Duncan, D. B. (1967). Estimation of the probability of an event as a function of several independent variables. *Biometrika*, *54*(1-2), 167-179.

Koh, K., Kim, S. J., & Boyd, S. (2007). An interior-point method for large-scale l1-regularized logistic regression. *Journal of Machine learning research*, *8*(Jul), 1519-1555.

Ng, A. Y. (2004, July). Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning* (p. 78). ACM.

Tibshirani, R. (2011). Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *73*(3), 273-282.

Goeman, J., Meijer, R., & Chaturvedi, N. (2012). L1 and L2 penalized regression models. *cran. r-project. or*.

Dean, J., & Ghemawat, S. (2010). MapReduce: a flexible data processing tool. *Communications of the ACM*, *53*(1), 72-77.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81-106.

Peng, W., Chen, J., & Zhou, H. (2009). An implementation of ID3-decision tree learning algorithm. *From web. arch. usyd. edu. au/wpeng/DecisionTree2. pdf Retrieved date: May*, *13*.

Xu, L., Liu, G., & Chen, Z. (2012, December). Research on optimization model of threshold setting for half-rate based on the decision tree algorithm. In *Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on*(pp. 316-320). IEEE.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *Journal of machine learning research*, *12*(Oct), 2825-2830.

*Decision Trees*. Online at <http://scikit-learn.org/stable/modules/tree.html> (referenced 19/08/2018).

Steinberg, D., & Colla, P. (2009). CART: classification and regression trees. *The top ten algorithms in data mining*, *9*, 179.

Rutkowski, L., Pietruczuk, L., Duda, P., & Jaworski, M. (2013). Decision trees for mining data streams based on the McDiarmid's bound. *IEEE Transactions on Knowledge and Data Engineering*, *25*(6), 1272-1279.

Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., & Wang, Z. (2007). A novel feature selection algorithm for text categorization. *Expert Systems with Applications*, *33*(1), 1-5.