Journal of Universal Computer Science, vol. 26, no. 1 (2020), 127-155
submitted: 30/12/2018, accepted: 15/11/2019, appeared: 28/1/2020 CC BY-ND 4.0

A Framework for Online Social Network Volatile Data Analysis: A Case for the Fast Fashion Industry

Anoud Bani Hani
(Zayed University, Dubai, UAE
anoud.bani-hani@zu.ac.ae)

Feras Al-Obeidat
(Zayed University, Abu Dhabi, UAE
feras.al-obeidat@zu.ac.ae)

Elhadj Benkhelifa
(Staffordshire University, Stoke-on-Trent, UK
e.benkhelifa@staffs.ac.uk)

Oluwasegun Adedugbe
(Staffordshire University, Stoke-on-Trent, UK
oluwasegun.adedugbe1@staffs.ac.uk)

Abstract: Consumer satisfaction is an important part for any business as it has been shown to be a major factor for consumer loyalty. Identifying satisfaction in products is also important as it allows businesses alter production plans based on the level of consumer satisfaction for a product. With consumer satisfaction data being very volatile for some products due to a short requirement period for such products, current consumer satisfaction must be identified within a shorter period before the data becomes obsolete. The fast fashion industry, which is part of the fashion industry, is adopted as a case study in this research. Unlike slow fashion, fast fashion products have short shelf lives and their retailers must be able to react swiftly to consumer demands. This research aims to investigate the effectiveness of current data mining techniques when used to identify consumer satisfaction towards fast fashion products. This is carried out by designing, implementing and testing a software artefact that utilises data mining techniques to obtain, validate and analyse fast fashion social data, sourced from Twitter, to identify consumer satisfaction towards specific product types. In addition, further analysis is carried out using a sentiment scaling method adapted to the characteristics of fast fashion.

Keywords: Sentiment Analysis, Sentiment Scaling, Data Mining, Social Mining, Consumer Satisfaction, Fast Fashion
Categories: K.4, K.7, K.8

1 Introduction

Consumer satisfaction is an important part of retail and business in general. Consumer satisfaction has been shown to be linked with consumer loyalty [Ali, Muqadas 2015]. As such, consumers who have positive experiences with a brand and product are more likely to purchase from that brand again. In addition, it has been shown that loyal customers are less price sensitive and less likely to purchase competitor services or products [Oliver 2014]. With the vast amount of data across social media platforms on the web and their exploration for several purposes such as opinion mining and sentiment analysis, it has become pertinent to explore and analyse these data for business growth across diverse industries. However, with some products and services having a very short life span, information published on social media platforms regarding such is very volatile, as it becomes obsolete very quickly. Due to this volatility, it is important to obtain and analyse the data within a short period and respond accordingly. While such data volatility can be observed across diverse sectors, the fashion industry is chosen for use as a case study in this research. Fashion is part of everyday lifestyle across different societal types and is a form of identity that socially projects parts of our culture and lifestyle [Paterson 2017]. Consumer satisfaction is of specific importance in the fashion industry as consumers must be satisfied enough to confidently wear the product and reflect that satisfaction to their peers. When consumers express their opinions publicly, they can influence other potential consumers either positively or negatively [Lea-Greenwood 2013].

With the massive growth of social media and its adoption by millions of people globally as part of their everyday life, it allows people to communicate and interact on a global scale. Interactions take place in various media forms from simple text communication to images, videos and more. The publicity of data on certain social media make them ideal for word of mouth by sharing opinions and experiences to peers. One of the primary social media applications that this research explores is Twitter. At the time of writing, Twitter alone has 126 million daily active users with an average of 500 million tweets sent per day [The Washington Post 2019]. A tweet is a short message of up to 280 characters in length, although it can also include media such as images and videos. With the real time nature of Twitter, it means that once a user posts a tweet, it is immediately public for other users (and potentially non-users) to see and interact with. The availability of this content serves as an invaluable source of information to businesses [Dhaoui et al. 2017].

However, the data is not guaranteed to be clean and there is high possibility that a sizable percentage of the collected data will be noisy data. This means that the data may have improper spelling and grammar [Joshi 2019] especially as the user must abide by the 140-character rule of Twitter. This could prove problematic when attempting to perform analytics on the data. The motivation of this research is to investigate how current social media mining techniques are used to extract and analyse meaningful data for the purpose of identifying consumer satisfaction in fashion retail. A critical analysis of these current techniques is presented to identify gaps in the process or areas that could be improved. The research will then focus on addressing these problems and creating a measurable solution.

The paper has been structured as follows; Section 1 introduces the report and the motivation behind it. Section 2 provides a context for the research, in terms of background information within the domain. Section 3 describes the solution proposed by this research while Section 4 analyses requirements for the proposed solution. Section 5 provides the design details of the solution while the implementation is described in Section 6. Section 7 describes the testing, including experimental evaluation results and a conclusion section follows thereafter.

2 Background

Fashion is a term often used to describe a product or market that it is short lived but also has a high demand [Cerruti et al. 2016, Madhani 2015]. While fashion can occur in many types of products and markets, the type of fashion being investigated in this research is that of clothing and accessories. As consumers wear these fashion products which are usually visible when being wore, they often associate fashion as being part of their identity [Arrigo 2016]. The fashion industry model has changed drastically where mass production is now becoming less common in favour of a new model whereby more ranges are produced but the lifespan of the range is decreased in order to meet consumer trends [Caro, Martínez-de-Albéniz 2015]. This new model has been coined 'Fast Fashion' and its benefits have been proven globally, making it the most recognised business model in the industry [Arrigo 2016]. Many mainstream retailers have already adopted the model in the UK fashion industry, and it is widely acknowledged as being a critical model for success in fashion retail [Caro, Martínez-de-Albéniz 2015].

The fast fashion model aims to become more adaptable to rapid change. As such a retailer must be able to acquire and put out new stock matching current demands and may need to mark down existing stock that is no longer required. For this reason, one of the principles behind the model is to reduce the amount of stock not on the shop floor at any one time. As stock is required in a just-in-time manner to match new trends, success can depend on time taken to design the product in a manner that incorporates consumers' needs, manufacture and get it to market [Cerruti et al. 2016]. Due to the volatility of fashion trends, if this process takes too long the product may no longer be in trend when it reaches the shop floor. In addition, retailers will need to achieve this before their competitors in order to gain a market advantage [Turker, Altuntas 2014].

The more the retailer meets consumer expectations, the more the brand will be able to build and sustain itself; bringing additional advantages such as increasing revenue and brand loyalty [Sreekumar, Rajmohan 2019]. As the retailer must adapt to consumers' expectations, [Cerruti et al. 2016] suggested that it should be a continuing priority to identify consumers shifting preferences and requirements. [Bhardwaj, Fairhurst 2010] also added that "by knowing how and to what extent rapid changing fashion affects consumers' purchase behaviour and satisfaction levels, retailers can develop strategies that can lead to improved profitability".

Despite the potential advantages of fast fashion, there is still a lack of academic literature on it [Caro, Martínez-de-Albéniz 2015] with majority of what is available failing to focus on the consumer [McNeill, Moore 2015] and how factors such as consumer satisfaction affects consumers' requirements. This research suggests that the fast fashion model is one of the main business strategies used in the UK fashion industry and has also attained global success. However most academic literature on fast fashion focus primarily on the supply chain process rather than from a consumer perspective. This has also been identified by [Caro, Martínez-de-Albéniz 2015], [McNeill, Moore 2015] and [Bhardwaj, Fairhurst 2010]. Identifying consumer satisfaction is important for adapting business strategies to maximise profits. It also provides a method of identifying consumer experience in relation to issues with products or services. Consumer satisfaction is also important as negative experiences may make consumers consider competitor products whereas positive consumer experiences help to build consumer loyalty and increase consumer retention. This leads to competitive advantages which are important for an already competitive market such as fashion.

However, with the speed of fast fashion, where products may quickly be in trend and then equally as quick be out of trend, consumer requirements can change rapidly. It is therefore important to be able to identify consumer trends in a short period of time while keeping up to date with them. The use of social media as data source appears to be a suitable one as it is constantly refreshed with new content posted by consumers, which can be accessed in real-time. To the best of the authors' knowledge, there is very little or no academic literature that explores the use of social media to identify consumer satisfaction within fashion. More specifically, how effectively the process can be carried out in a fast fashion environment, where changes occurs very frequently. From the review of social mining techniques for social media data and the performance of multiple types of analysis in order to identify consumer sentiment and satisfaction, one of the main challenges will be to evaluate whether these techniques are suitable and effective when used in fast fashion where consumer requirements and products are likely to change within short periods.

Fast fashion retail outlets can often employ a 'here today, gone tomorrow' mentality whereby multiple lines of products are available at any one time but with different shelf life spans. This diversity of products may make it difficult to differentiate between them when trying to identify consumers' satisfaction. Therefore, the second challenge will be to examine the effectiveness of social mining techniques for identifying consumer satisfaction towards specific fashion ranges. The third challenge would be to determine whether the existing social mining techniques could be tailored to better suit the fashion environment if inadequacies are identified. If so, it would also be worthwhile in a future project to identify whether these modifications could prove beneficial in other industries with similar environmental features and nature.

This research also revealed a common theme whereby fast fashion consumers expect a lower quality product as they are purchasing a cheaper product in comparison to slow fashion whereby consumers expect a quality product as it costs more. Due to the expected lower quality, consumers' expectations are equally lower. As such, fast fashion consumers experience more expectation confirmation if a product is damaged within a short period of time with the level of satisfaction increasing if the period is longer. Other factors, including initial cost of the product and how many times it was consumed are also variables. Standard methods of sentiment analysis may not take these secondary factors into account.

3 The Proposed Solution

The proposed solution to the identified gaps is to design and implement a software artefact that can use social media as a means of identifying consumer satisfaction in this industry. The implementation shall initially use generalised social mining techniques in order to obtain and mine the required data. As part of the solution, an alternative method tailored more towards fashion is proposed, designed and implemented. These can then be compared to see if the latter provides any advantages over the former. The process flow of the solution comprises of four major phases, as represented in Figure 1 with a brief description of each phase following.

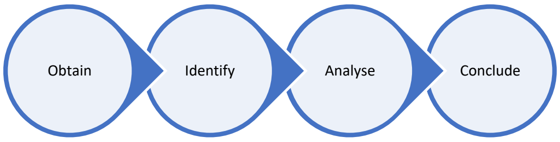

Figure 1: The Proposed Solution Process Flow

1. Obtain: The solution will need to obtain appropriate social media data from a social media application.
2. Identify: The solution will need to identify the relevance of obtained data. To optimise resources and improve results, it is important that irrelevant data is not analysed and therefore social mining techniques will be required to validate each piece of data.
3. Analyse: The solution will need to analyse the relevant data to determine consumer sentiment and satisfaction. As the proposed solution will have identified the current data as being important, each piece of data will need to be analysed to help form the conclusion in the fourth phase.
4. Conclude: The solution will need to be able to draw conclusions based upon the collective analysis. The solution will need to produce quantitative data displaying the measured satisfaction and related data.

During the second and third phases, it will be important to investigate whether general social mining techniques will be effective when considering fashion-oriented data. The fourth phase will need to produce quantitative data. By storing the dataset used, it can be analysed using other methods (such as manual review) to compare the effectiveness of the solution and to test the suitability of using social media to identify consumer satisfaction in fashion retail. The proposed solution to tailoring the sentiment analysis process would be to design a method whereby dynamic rules can be set depending on the topic of the process.

In this instance, one rule would be to dynamically scale the weighting of the sentiment based upon identified contexts. An example of this would be scaling down negative sentiment in ratio to the length of time the product has been owned where damage and time contexts are identified. Rules such as this may help to account for characteristics of the fast fashion industry. The process would then need to be tested to evaluate its benefits, and if so, the process may also be applicable to other industries and sectors outside of fast fashion.

4 Requirements Analysis

Prior to commencing design of the proposed solution, it is pertinent to analyse its requirements and constraints to determine the expected capabilities. One of the main challenges identified from the gap analysis is to test whether existing data mining techniques could be used to identify consumer sentiment towards specific products in the fast fashion industry. To implement this, a data source would be required for the data mining techniques. These data would need to contain consumer’s opinions on products for them to be relevant. With this research investigating different methods for gathering these data, the findings suggest that social media would be the most appropriate option as data is consistently being generated by potential consumers and in large quantity. Due to the nature of social media, it would also be accessible autonomously as such data is in the public domain. Based on the amount of data that will be available, a verification process would be required for the data to prevent large amounts of resources being utilised to analyse irrelevant data.

Once verified, the solution will need to identify products being mentioned in the data. This could be carried out using Named Entity Recognition, the process of identifying entities within data. Once products have been identified, the solution will need to calculate the sentiment towards each product within the data. One of the other challenges that was identified from the gap analysis was whether this could be tailored towards fast fashion. A potential solution that was identified for this would be to allow the calculated sentiment to scale based on the context present in the data and how it relates to the characteristics of fast fashion. The application will therefore need a method of obtaining this scaled sentiment. Once this has been obtained, the analysed data can be reviewed. A summarised breakdown of requirements that will need to be met by the proposed solution is as follows:

- The solution will need to be able to obtain relevant data from a social network source.
- The solution will need to be able to identify products being mentioned in the obtained data.
- The solution will need to be able to perform sentiment analysis on the obtained data.
- The solution will need to be able to scale the performed sentiment index according to characteristics common to fast fashion.
- The solution will need to be able to collate results from analysed data to produce an overview of the consumer sentiment index for identified products.
- The solution will need to store obtained and analysed data for potential further research and analysis.
- The solution must be suitable for the characteristics of fast fashion.
- The solution will comprise of four primary system modules. These are Data Retrieval, Validation, Analysis and Review modules. Other modules will include the database system for storing data produced by each of the primary modules.
- Interaction between the primary modules is represented with Figure 2.

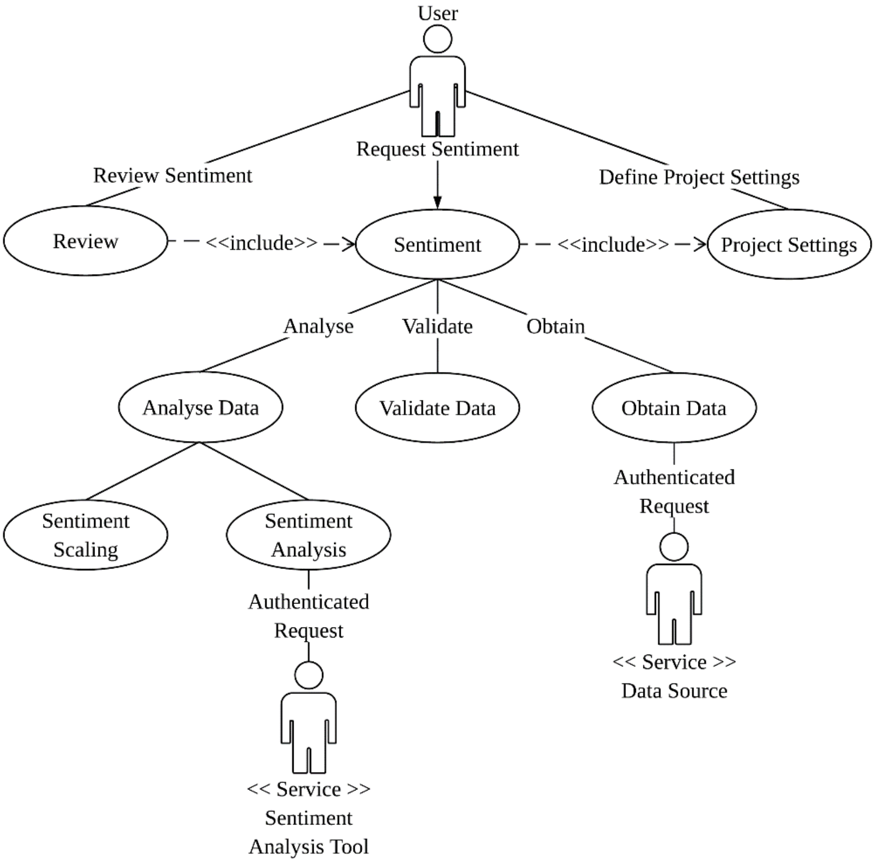

Figure 2: Interaction representation between the primary system modules

Furthermore, these requirements provide a basis for mapping out functionalities towards utilising the proposed solution and functionalities that the solution will facilitate. Intended users of the solution will be individual fast fashion companies that require it to analyse consumer sentiments aimed at their own products rather than fast fashion collectively. The solution will be designed to be modular. This will enable each module to be implemented and tested independently. From a software development standpoint, it also fosters maintainability of the solution as each module will be specialised and independent of the other modules. While any module may use data from a previous one, they will not be dependent on the previous module. Figure 3 further presents a high-level flow of the proposed solution with functionalities split into the four core modules (Obtain, Validate, Analyse and Review) defined earlier.

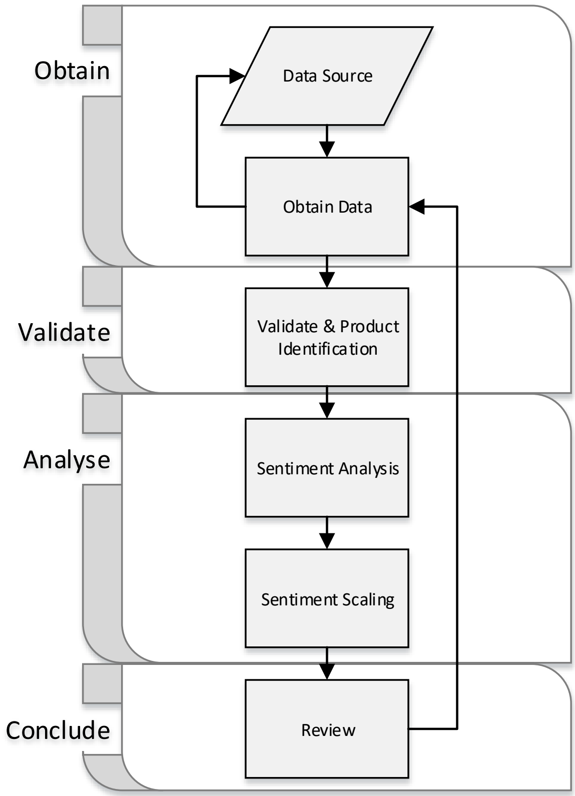

Figure 3: High-Level Flow of the Proposed Solution

5 The Solution Design

As stated in the requirements analysis, the proposed design consists of four primary modules which also defines stages for the solution design. The design for each of these is described in this section.

5.1 The Obtain Stage

The purpose of this stage is to obtain relevant social media data from the chosen data source. This is data that will be analysed by the application in subsequent stages. The data obtained will need to be relevant to the fast fashion industry. There are two potential approaches for the type of data that could be obtained. Generalised fast fashion social media data could be obtained to identify current trends within fast fashion. Alternatively, fast fashion social media data could be collected for brands in which case it will be possible to identify current trends specific to the chosen brands. Twitter's search endpoint returns tweets that have been posted from the time of the query to between six to nine days ago. This implies that tweets older than six to nine days may not be returned. While this may pose an issue for most industries, research into characteristics of fast fashion shows that is very much a 'here today, gone tomorrow' type of industry. As fast fashion is fast paced, social media data will also be outdated faster than that of a slower industry. For this reason, this limitation of Twitter does not pose as much of a disadvantage for the industry.

The first time that the application is run, it will attempt to obtain all available tweets. Once all available tweets have been obtained, the highest obtained tweet id will be saved as new tweets will have a higher id. On subsequent runs the application will obtain tweets until either a tweet has been found that is equal to or lower than the last obtained id or until all available data has been processed, whichever comes first. This will ensure that all available new data not previously obtained will be obtained and processed accordingly. Figure 4 presents a flowchart which describes the process further.

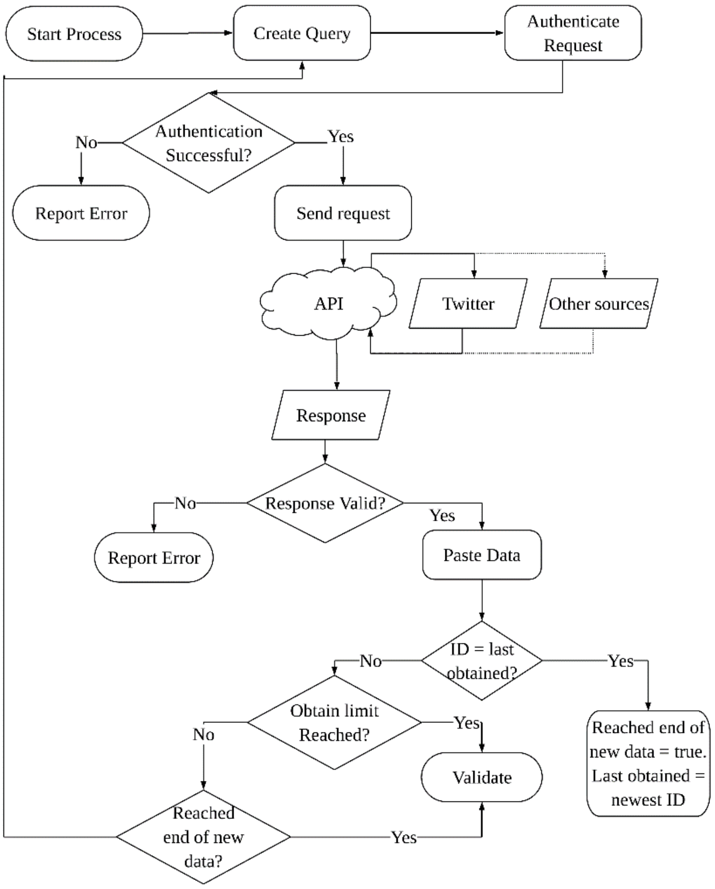

Figure 4: The Obtain Module Process Flow

In order to use the Twitter Search API, queries must contain appropriate authorisation headers. For this, Twitter uses OAuth authentication. The Twitter Search

API supports two forms of authenticated query. The first is on behalf of an application whereby only public data can be retrieved. The second is on behalf on a user whereby public data and data specific to that user can be retrieved. The application being developed will only need to access public data as it will not be used to manage or on behalf of a specific Twitter user. This also implies that the application does not require authorisation from the user. Instead it will obtain a 'Bearer Token' from Twitter which can be used to authorise queries. This will also subject the application to only being able to make 450 queries every fifteen minutes. Figure 5 presents a process flow for Twitter bearer token OAuth.

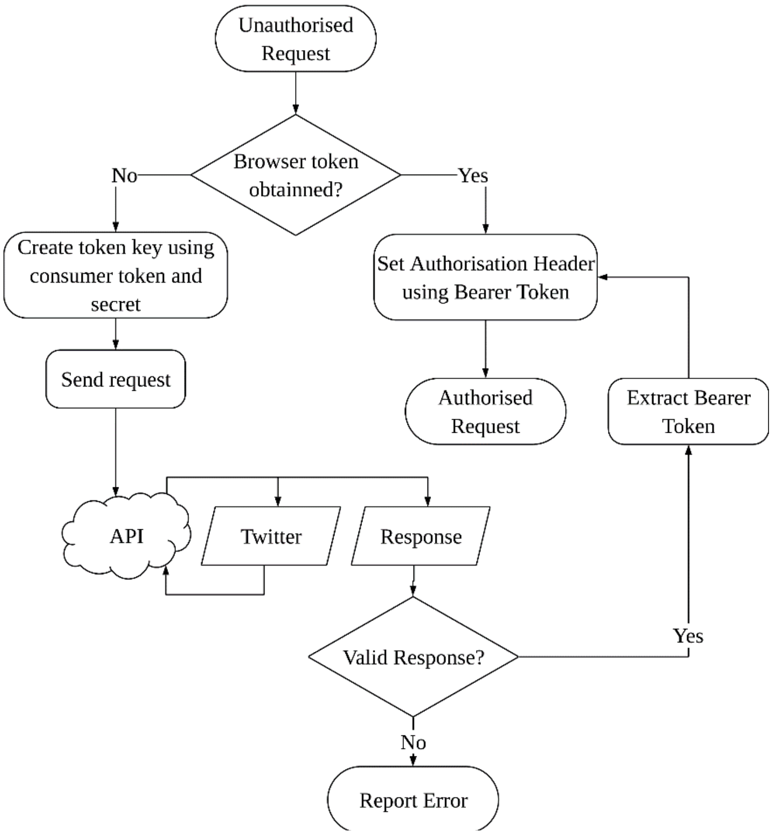

Figure 5: Twitter Bearer Token OAuth Flow Diagram

Twitter also offers a streaming API. The streaming API allows streaming connections to be established with new tweets matching the set criteria being pushed to the requester as they are posted. However, it is not appropriate for use in this context as initial investigations reveal that data for several fast fashion retailers is being produced slower than what can already be obtained using the Search API (100 tweets per two seconds). If the application has not been running constantly, it will also be important for the application to gather tweets for up to three weeks before the date it was used again, which is not possible through the streaming API.

5.2 The Validate Stage

The second stage of the application will be to carry out validation on the social media data collected in the previous stage. This is to ensure that time and processing during the analysis stage is only carried out on relevant data. What makes data relevant may change depending on the aims of the application. For this reason, the process of performing validation must be flexible enough to allow new validation checks to be added or removed as required. In this context, validation will be performed based on characteristics of and for use in fast fashion. There will be two main types of validation checks carried out on tweets; relevancy and authenticity. Relevancy checks will be carried out to ensure that the data is relevant to the research area. In this case, it will ensure that the data is relevant to fast fashion. One of the main reasons for carrying out relevancy checks is that retailer's names can also have other meanings that are well used. An example is Zara, which is one of the biggest fast fashion retailers. Zara is also a name and a standard search for Zara would also return tweets from users with the name Zara or with references to a person named Zara. While searching for tweets containing @Zara, the relevancy check would ensure that tweets from people mentioning Zara without any links to the official Zara account would not be retrieved. The relevancy checks will therefore attempt to identify that the data is referring to the chosen retailer and that it is about products you would expect from that store.

Furthermore, the relevancy validation would include product identification. One of the core purposes of the application is to identify consumer sentiments towards specific product types. For this to be possible, information relevant to a product type must be included within the data. Without this, it may not be possible to identify such products. Therefore, if this information is not identifiable within the data, the application will have no purpose to conduct a sentiment analysis on it.

In addition, product identification will use semantic role labelling by identifying keywords from a pre-defined set. The set of words will correspond to either a product type, such as shirt or dress, or it will correspond to a product attribute such as colour or style. These will be common to Fast Fashion retailers. The validation will require at least one product type to be present although multiple product attributes may be present. The product types and attributes found will be recorded and linked during the validation stage. Figure 6 presents the overall process flow of Relevancy Checking.

The Authenticity checks will also be carried out to ensure that gathered data is from consumers. This is because there may be data from other retailers mentioning or selling products from the retailer being analysed. When performing a search on leading fashion retailers such as 'River Island' and 'Dorothy Perkins', there were several tweets in both searches advertising eBay listings for products from these retailers. The application will therefore fail any data containing external links. This is because the application's purpose is to obtain consumer sentiment whereas advertisements are likely to contain little sentiment which potentially could neutralise results.

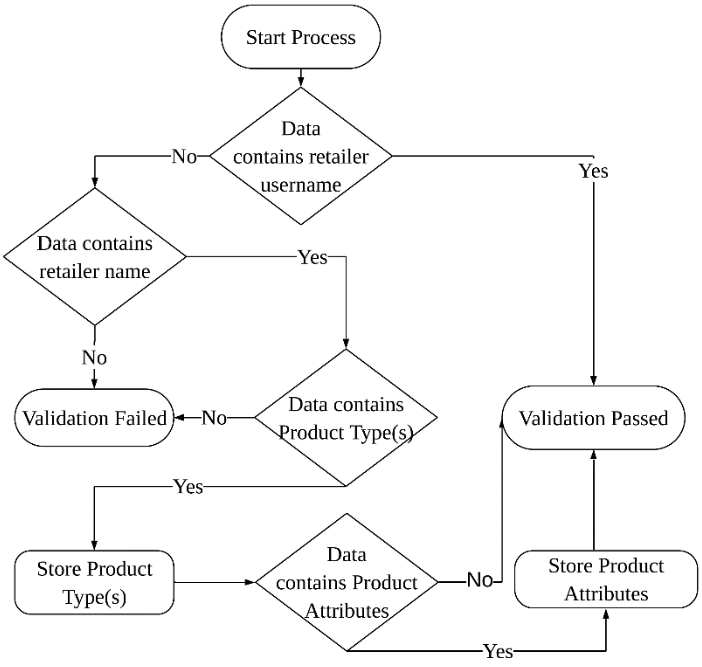
Start Process
Data contains retailer username
No
Yes
Data contains retailer name
Yes
No
Validation Failed
No
Data contains Product Type(s)
Validation Passed
Yes
Store Product Type(s)
Data contains Product Attributes
No
Store Product Attributes
Yes

Figure 6: Overall Process Flow of Relevancy Checking

5.3 The Analyse Stage

This stage implements sentiment analysis for data that passed validation. Firstly, base sentiment scores will be generated for the data. Once these have been obtained, the scores will be scaled to match characteristics of fast fashion. Such characteristics include the fact that fast fashion products tend to be lower in cost and quality. The research investigation revealed that fast fashion products are therefore not expected to last as much as more expensive slow fashion products. In this stage, sentiment scaling would work by scaling the negative sentiment down based upon the amount of time that has passed since purchased, providing that this data is available. The first task of the sentiment scaling would be to check for context within the data suggesting 'damage' to the product being mentioned. If found, a quantity and a cause will also need to be found. In this case, the cause will be 'washed' and the quantity would be the amount of times it had been washed. If this is also found then the sentiment will be scaled according to a defined range whereby the more the product has been washed, the more the negativity score will be scaled down. If there is not enough context data present, sentiment scaling will not be performed. The initial design for this stage was to extend an external sentiment analysis API. A third-party external API known as Skyttle was chosen. Skyttle is a web-based service that carries out phrase-level sentiment analysis on up to 10,000 characters of text per API call. It then returns the confidence of the text being positive, neutral or negative. Skyttle was chosen over other potential sentiment analysis tools as it performs the task required and is also more affordable.

The initial plan was to create a custom API that extended Skyttle API in the following way: Firstly, it would use Skyttle API to obtain sentiment scores for the received text. It would then extend this by performing sentiment scaling on retrieved values which would then get returned in addition to original values. A key advantage of this would be that other applications could have made use of the added functionality by calling the custom API. Figure 7 presents this initial design.

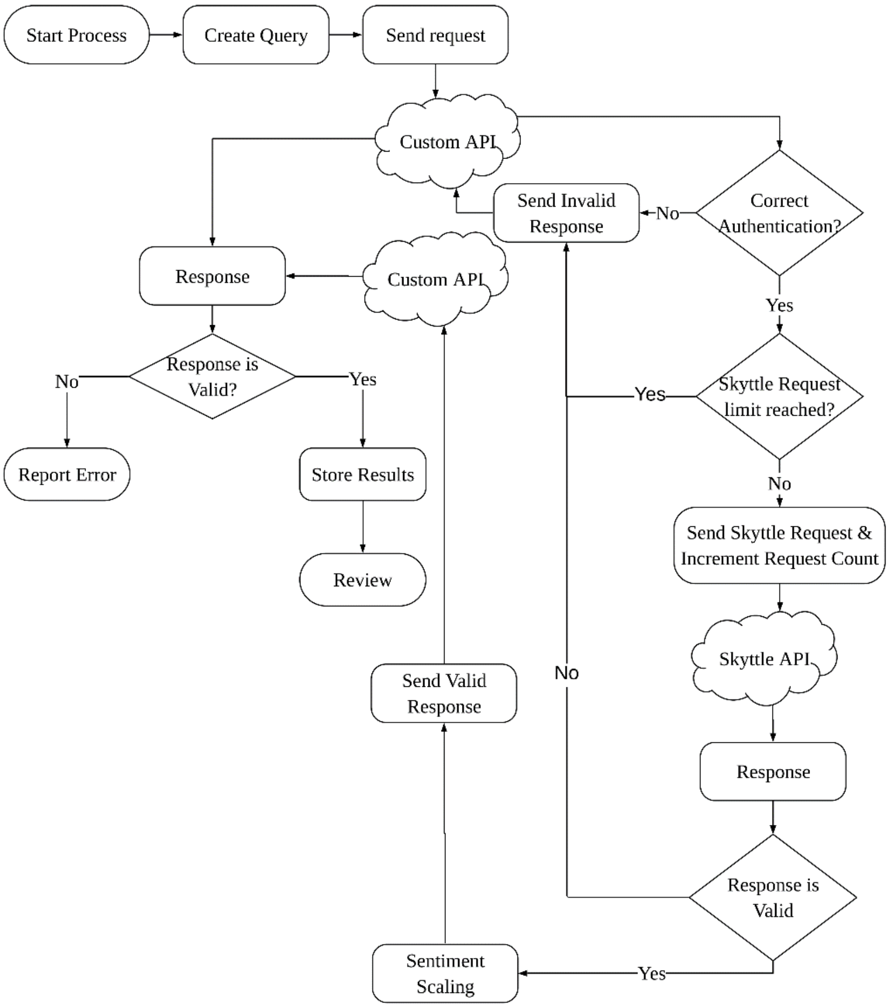

Figure 7: Initial Analysis Process Flow

During the testing stage it was also noted that Skyttle was sometimes unable to correctly identify consumer sentiment for some basic phrases that should not have been difficult to determine. An example of this is that it identified the word 'love' as being neutral. In turn a phrase such as 'I love my new scarf!' would also receive neutral consumer sentiment index whereas this statement is clearly positive. As Skyttle is not open source, it was not possible to observe what may be causing this. To resolve issues found with Skyttle, a novel sentiment analysis technique was implemented which will imply carrying out the process within the application. While this new implementation approach prevents re-use from other applications, it also brings several key advantages. Firstly, there will no longer be an API limit, which implies that analysis can be continuous for all obtained data. Secondly, with access to the API codebase, it can be improved over time and extended when required. Thirdly, an internet connection is not required causing no issues that can result from connection failures. The sentiment analysis will be based on lexicon methods whereby a corpus of key terms and their sentiment weights are defined [Muhammad et al. 2016]. The application will use key term recognition to assign appropriate weight to data. The analysis will also be carried out on tweets individually rather than collectively. This is like the approach carried out by [Paltoglou, Thelwall 2012]. For each tweet a base sentiment value will be calculated using the defined lexicon. Figure 8 presents the base sentiment calculation flow.

This works by matching keywords and assigning each word the value defined in the lexicon. A positive (+) value implies that the sentiment is positive whereas a negative value (-) means the word is negative. If the word is not found in the lexicon, it is assumed to be neutral. Intensifier words are also identified. If found before a non-neutral sentiment word, the sentiment value for that word would be multiplied by the intensifier value. Finally, negation words (¬) such as 'not' will be identified. These will reverse the sentiment of non-neutral words. An example of how this works is displayed in Table 1.

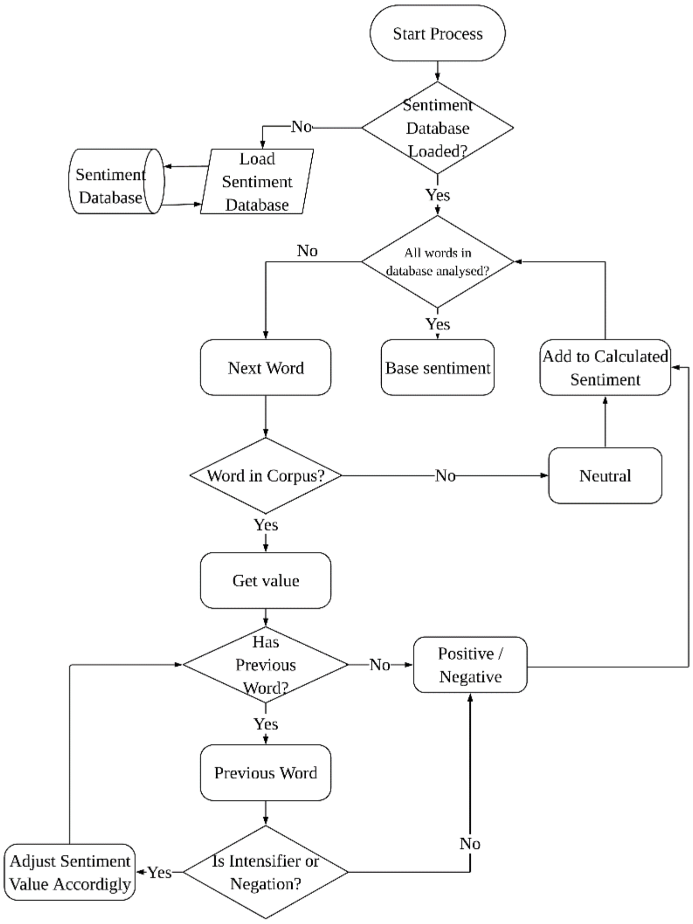

Figure 8: Base Sentiment Calculation Flow

Neutral	Neutral	Negation	Intensifier	Positive	Neutral	Neutral
0	0	¬	1.5	2	0	0
I	am	**not**	**really**	**liking**	this	scarf.

Table 1: Sentiment Scoring Example

From Table 1, the phrase ‘I am not really liking this scarf’ contains a negation, intensifier and positive sentiment word. This calculates to ¬ (1.5 x 2) = -3. If the sum is above zero, the phrase is classed as positive whereas if it is below zero, it is classed as negative. The distance from zero represents how positive or negative the phrase is. Figure 9 presents the sentiment scaling flow diagram.

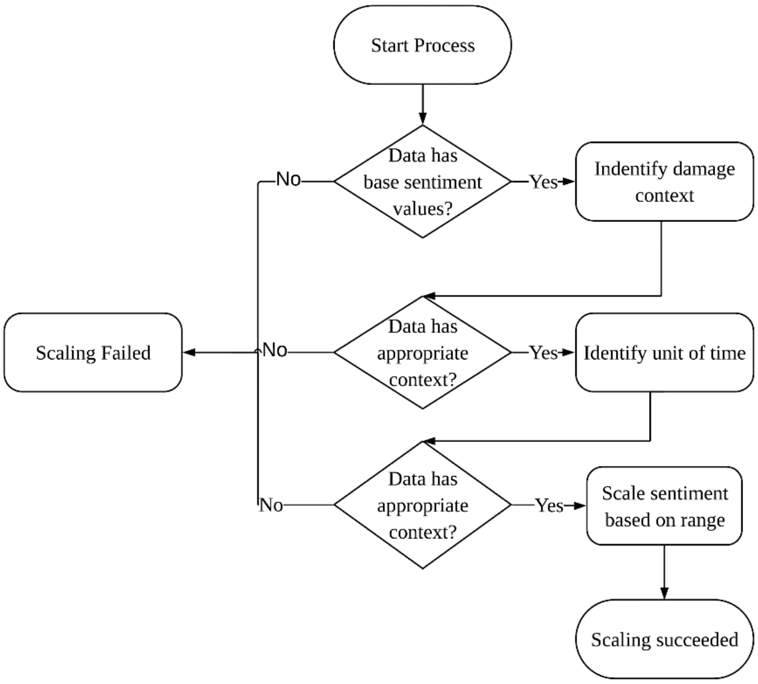
Start Process
Data has base sentiment values?
No
Yes
Indentify damage context
Scaling Failed
No
Data has appropriate context?
Yes
Identify unit of time
No
Data has appropriate context?
Yes
Scale sentiment based on range
Scaling succeeded

Figure 9: Sentiment Scaling Flow Diagram

Once the base sentiment has been calculated, it will be scaled accordingly. Firstly, key term identification will be used to identify context present in the data. If the context is not present, scaling cannot be performed. On the other hand, if it is, the application will identify a measurement such as an amount of time. Due to the characteristics of fast fashion, products are not expected to last for so long. As such, data in which time and damage context has been identified will be scaled so that the negative sentiment is reduced in ratio to the greater amount of time measured. The scaling will be capped so that it will not be possible for a negative statement to have zero negativity, instead it will have a reduced negative weight rather than zero.

5.4 The Review Stage

The ‘Review’ stage will query all analysed data currently stored to extract information such as products types currently receiving the highest index of positive or negative consumer sentiments. The reviews will also contain related attributes of product types and their sentiment indexes. There will be three types of reviews: core, attribute and full types. A core type review will comprise of all product types and their related sentiment indexes. These will provide a broad overview of which categories of product are currently held with positive sentiments or may currently have negative sentiments. The second type of review will be an attribute review. This will be like the core type review except for the fact that it will only focus on attributes such as colours and styles. They will not be linked to specific categories of products but will allow users see which attributes are currently being highly regarded. The final type of review will be a full review. This will be for all products based on their sentiment and include sentiment indexes for attributes found for each product, enabling users see which attributes are currently held with positive or negative sentiment indexes for each product type. This information can then be utilised by the retailer to decide which attributes to put more focus on for design and distribution. Figure 10 presents the full sentiment review process flow.

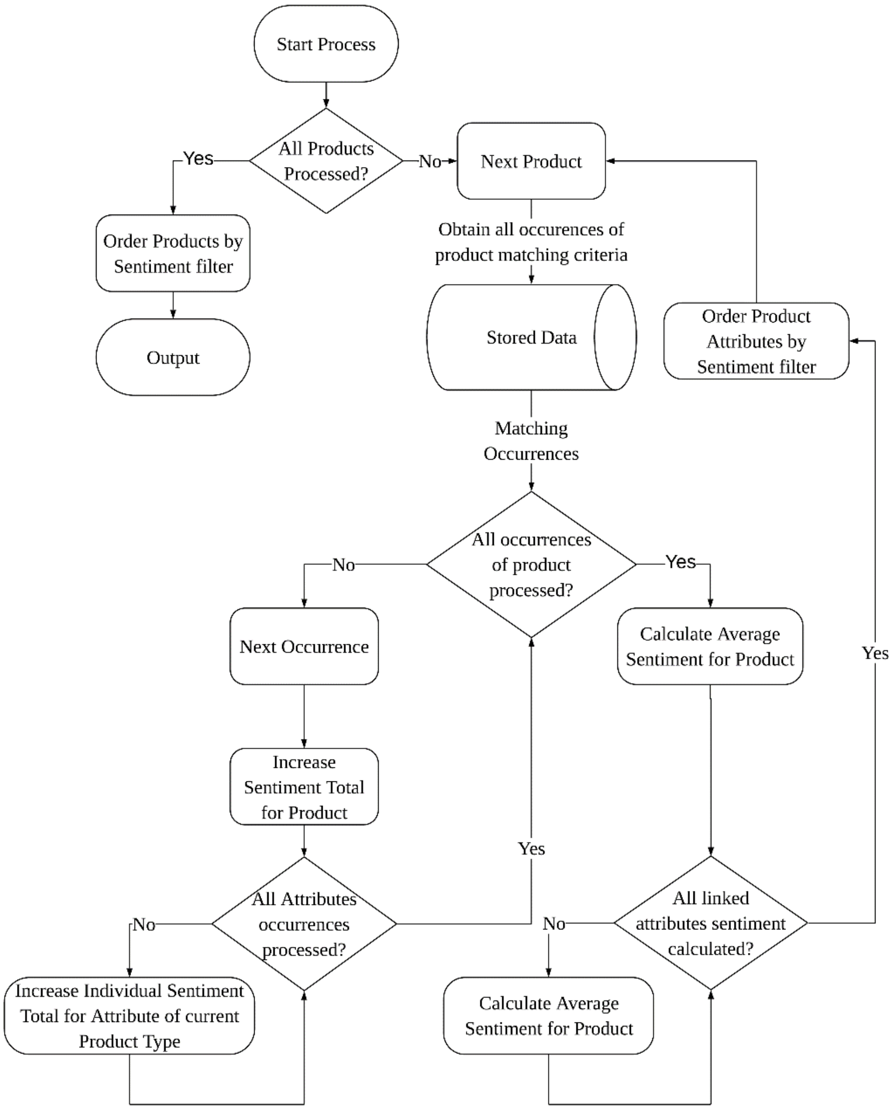
Start Process
All Products Processed?
Yes
No
Next Product
Order Products by Sentiment filter
Output
Obtain all occurences of product matching criteria
Stored Data
Order Product Attributes by Sentiment filter
Matching Occurrences
All occurrences of product processed?
No
Yes
Next Occurrence
Calculate Average Sentiment for Product
Yes
Increase Sentiment Total for Product
Yes
All Attributes occurrences processed?
No
No
All linked attributes sentiment calculated?
Increase Individual Sentiment Total for Attribute of current Product Type
Calculate Average Sentiment for Product

Figure 10: Full Sentiment Review Process Flow

6 Implementation

This section describes the implementation process for the application, including challenges and decisions made. It also includes changes that had to be made to the original designs.

6.1 The Database

An SQLite database was implemented to store application data. This choice is because it can be created and managed directly within the application, local to the machine. The database is not complex and will only need to store up to three weeks' worth of data at any given time. It is managed by a class called 'ProjectDatabase'. This is a class specialised for handling data in the format of tables in the database. It takes out the complexity of having each module requiring interaction with the database directly by making their own SQL queries. It also helps to prevent any SQL errors as the database manager is more able to validate data due to knowledge regarding data to be inputted. Another benefit is that it decouples the database implementation from each module. Each module can focus purely on handling the data for its specific task and pass such data on to be handled by the database. While the singleton pattern has not been directly applied, the application will only have one instance of the main database manager class. This ensures that all calls are made onto the same database. If complex multithreading was utilised, this would also help to set up synchronisation to ensure that multiple transactions can be made safely. The 'ProjectDatabase' class is a subclass of a general 'DatabaseManager' class. This has been done to improve code re-use should another database need to be added in the future either to this application or within an external one. Only the 'ProjectDatabase' class has implementations specific to this application. Another database present in the application is the Sentiment database. This is a small basic database that serves to store and access consumer sentiment definitions and indexes loaded from corpus documents included in the implementation. This database is not specific to individual applications, so it has been separated from the application database.

6.2 The Data Types

One change to the initial designs during implementation was the decision to include a 'DataTypes' namespace containing structures that can be used across all the different modules. The initial design was to use data structures unique to each module, and then use a mixture of common data types and convertors to communicate the data between the modules. By using converter objects, each core module would be decoupled from others. While this was working well during the first incremental stages of the implementation, issues began to occur. One issue was performance hits that occurred when needing to communicate data between the modules. These were caused when the application needed to convert data of one type produced by a module to another format for use with the database and then for use with the next module.

Another issue was that the code was starting to become overly complex and harder to read. One reason for this was the database module. As this needed data input from all modules, it required several different output and input types. To prevent class explosion and several classes identical to the classes in the external modules, an attempt was made to use common data types provided by the .NET framework. The complexity increased to a point where structures such as lists were within nested dictionaries. The validation of this data became even more complex with checks required for the correct keys. Without this additional validation the application would need to assume the keys exist which could prove problematic when maintaining the code in the future.

The solution was to create the new namespace consisting of structure objects that can be used across all modules. These structures serve no other purpose than to contain data, and as such perform no logic on the contained data. The immediate benefit was improved clarity of code. As each structure consisted of specific named properties for datasets the structure was designed for, it was easy to observe data required and that will be available. The need for additional validation was also removed as the properties will always exist even though property values may be null. As the structures can be persisted between modules, the need for conversion was also reduced, improving application performance.

6.3 The Obtain Module

The Obtain module gathers data from Twitter (the chosen data source) and then parses the required data into a suitable object format. The main class of the Obtain module is the 'TwitterWorker'. This is the class that needs to be called in order to carry out the maintain functionality. The 'TwitterWorker' has three different events that are triggered at defined times while carrying out its' obtain functionality. These are 'TweetsObtained', 'Started' and 'Finished'. The 'Started' event is triggered when the obtain loop is started and the 'Finished' event is triggered when the obtain loop is finished whether due to completion or due to error. The 'TweetsObtained' event is triggered each time a set of tweets have been obtained and parsed successfully.

Events have been used to decouple the module from the rest of the application. Events allow the application to notify external classes that have registered to listen to those events without the Obtain module needing to know anything about the listening objects. Another benefit is that the module can alert listeners multiple times through a single run of the Obtain stage. This is unlike if the method was instead set to simply return data as the method could only return once it has ended. This means other classes (including the one controlling the UI) would have no idea about the progress of the function. There were a couple of decisions that needed to be made during the implementation of the Obtain stage. Firstly, the number of tweets that should be obtained per API call. The maximum allowed (100 tweets) was used rather than the default (15 tweets) as this would allow more tweets to be processed in a shorter amount of time. It would also result in less API calls being made, helping the application to stay within its limit of 450 Search endpoint calls every 15 minutes.

Another choice that had to be made was how many tweets should be collected before completion of the Obtain stage. The application was designed to continuously collect new tweets until all available tweets have been reached or until the application encounters a tweet it has previously obtained during an earlier run (whichever happens first). The advantage of this is that the first run of the Obtain method can obtain all available data for the given search term, whereas subsequent runs will only obtain new tweets. By recording the highest obtained tweet id and persisting it across multiple runs of the application, it also ensures that only the latest tweet data is obtained, and API calls are not wasted gathering previously obtained data.

6.4 The Validate Module

In similar fashion to the Obtain module, the Validation module has a primary worker class called 'ValidationWorker'. This is the main class that should be used to perform validation. It also has three primary events: 'DataValidated', 'Started' and 'Finished'. These work in the same way as those mentioned in the Obtain module and provide same benefits too. Each dataset that is passed into the Validation module undergoes a series of validation tasks. There were several considerations that needed to be made for this. The first consideration was how the validation tasks were going to be implemented. A 'ValidationTask' interface was created which all further validation task classes implement. This interface contains the single method that should be performed by each validation task – validate. Using an interface brings three key advantages. Firstly, it ensures that all validation tasks perform that critical piece of functionality and return the required data. Another advantage is that instead of having to call each validation task separately, they can be grouped into a collection of 'ValidationTask' objects. As they share a common interface, the implementation does not need to explicitly call each individual validation task. Instead it can iterate through the list calling just the common functionality until all validation tasks are complete or until a validation task fails (whichever occurs first per piece of data). The final advantage is that new validation tasks can easily be added to the application by implementing the interface and adding it to the list. This makes the module easy to maintain.

Another decision was whether to return all validated data regardless of validation outcome or whether to return only the data that passed validation. A decision was made to allow for all data to be returned. While only data that passes validation will continue to the Analysis stage, the application will still need to know which data was validated to remove it from the database. As the full dataset is returned, it gives more control to the listening objects on how they wish to handle the data.

6.5 The Review Module

Unlike previous modules, the Review module does not output data and therefor has no events. One key decision that was made during implementation was to remove the review screen and replace it with its own window. The main reason for this was to prevent user interface complexity from having the review selection underneath a toolbar and a menu bar. It also allows users more resizing options over the main window to be able to view more information on the screen. Figure 11 presents a screenshot of the Review screen design.

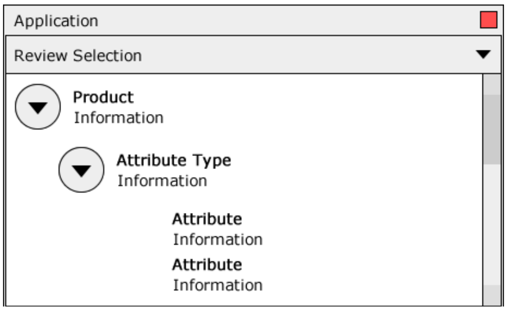

Figure 11: The Review Screen Design

Information within the review screen is listed as review items. These follow the same structure regardless of whether they are products, attribute types or attributes. Review items can be nested allowing for a tree of attribute types and attributes or products, attribute types and attributes. This provides a uniform method of outputting the different levels of detail created by the different review types.

7 Testing and Evaluation

This section details the testing stages that were carried out on the implementation. These include the experimental tests that were carried out in order to obtain the qualitative data that was used to answer questions set out in the gap analysis. Throughout the development, unit testing was performed on the implementation to ensure that each module worked correctly under various conditions.

7.1 Experiment Designs

In order to evaluate the accuracy and potential effectiveness of the application, multiple tests will be carried out. This section details each of these tests including their implementation procedure and purpose for each test. To ensure consistency between tests, same fast fashion retailers will be utilised as test subjects. Four retailers have been selected for this purpose; Zara, River Island, Dorothy Perkins and New Look. While the selection was done randomly, it provides a strong technical exposition as the brands are within the fast fashion industry and sufficient social media data was obtained for the experimentations. The first test will investigate the Obtain module of the application. Of keen interest will be investigating the amount of data that can be collected on the first run of the application per retailer and the time it takes to complete the tasks. This will enable the size of the dataset that is obtained to be observed and will be used in the subsequent modules. This is likely to vary based on size of company. If this module returns little data, then the process will not be very effective.

The second test will be examining the Validation module. Firstly, investigations will be carried out into the accuracy of the module. This will be implemented by taking a random sample of 1000 obtained tweets across a few retailers. These tweets will be manually validated to identify expected results of the validation stage for each one. Results obtained will then be compared to the actual result generated by the application.

This will enable the accuracy to be determined as the amount of results that were calculated as expected divided by the total amount of results (1000). For instance, if 100 of the 1000 tweets produced unexpected results, this would produce an accuracy of 90% (900/1000). The time taken to perform the validation will also be measured.

The third test will investigate accuracy of the Sentiment Analysis module. As a lexicon-based approach is being utilised, it is important to ensure that it correctly weighs identified key terms and applies the correct modifications based on identified intensifiers. In order to test the accuracy of the Sentiment Analysis module, a random sample of 1000 validated tweets will be manually analysed to determine whether the expected output should be very positive (above 3), positive (1 to 3), neutral (0), negative (-1 to -3) or very negative (below -3) . This is like the approach taken by [Paltoglou, Thelwall 2012]. The accuracy will be measured by dividing the amount of results that were calculated as expected by the total amount of results.

While sentiment analysis will be performed on individual tweets based on findings from research, an interesting test will be to investigate the potential advantages or disadvantages of performing sentiment analysis on documents consisting of multiple tweets. To do this, the Sentiment Analysis module will be tested using multiple document sizes ranging based on an incremental number of tweets. The two key factors that will be measured in these tests will be the speed of the analysis and the accuracy at which multiple tweets can be analysed as a single document. The final test will be to measure the accuracy of the Sentiment Scaling process based on defined ranges. This will be tested in a similar fashion to previous tests with 1000 random data samples being used. In addition to this, any data that is scaled will also be reviewed for accuracy.

7.2 Experimental Results

From the experiments conducted based on the four randomly selected retailers, Table 2 contains statistical data of the number of tweets that were obtained, validated and which had enough context to perform scaling for each retailer.

Retailer	Obtained	Validated	Scaled
Zara	114892	14925	12
New Look	62196	287	2
River Island	8490	127	2
Dorothy Perkins	2651	80	0

Table 2: Initial Results

Table 3 presents data from a second run of the application that took place several hours after the initial test. Both tests were conducted from scratch gathering the entire available Twitter data for each retailer query at the time of testing.

Retailer	Obtained	Validated	Scaled
Zara	117703	15188	12
New Look	66296	277	2
River Island	8162	103	1
Dorothy Perkins	2343	86	0

Table 3: Secondary Results

The data in Table 4 presents statistics about the Obtain module of the application across both runs. It is immediately clear from the results that a vastly different number of tweets were available for each retailer. This was expected as both 'Zara' and 'New Look' are words/phrases that are more likely to appear in other contexts. Zara and New Look are also larger fast fashion chains. As per Test One, the results show that tweet data was obtained at an average of 41 tweets per second. This includes time taken to parse and store the data. At an average of 82 tweets per 2 seconds, this is running at 82% efficiency in comparison to the 100 tweets every 2 seconds rate limit that Twitter applies.

	Run 1			Run 2		
	O	T	APS	O	T	APS
Zara	114892	46:56.867	40.79	117703	48:38.666	40.33
New Look	62196	25:19.187	40.94	66296	26:59.123	40.95
River Island	8490	03:22.3	41.97	8162	03:15.844	41.68
Dorothy Perkins	2651	01:04.318	41.22	2343	00:58.331	40.17

O = Obtained, T = Time Taken (MM:SS) and APS = Average amount validated Per Second.

Table 4: Obtain Module Results

Tables 5 and 6 display results for the Validation module over the two runs. On average across all four retailers only 5.26% of obtained tweets passed validation. From the observation of individual cases, this ranged from as high as 12.95% (Zara) to as low as 0.13% (New Look). Except for Zara that had an average of 15,057 tweets passing validation, the other three retailers had an insignificant amount. As the obtained data spans 6 to 8 days, this is an insufficient amount of data to represent overall consumer sentiment.

In the second run of the application, the validation was instead timed performing the raw validation and not updating the database. Similar results were identified meaning that the validation did not lose efficiency. However, the average number of tweets validated per second increased by 3,775% on average. This shows a possible area of optimisation for how validation is conducted. This is further presented in Figure 12.

	P		F		T	APS
	#	%	#	%		
Zara	14925	12.99	99967	83.01	21:13.947	90.186
New Look	80	0.129	62116	99.871	07:38.988	135.507
River Island	287	3.38	8203	96.62	01:31.582	92.704
Dorothy Perkins	127	4.791	2524	95.108	00:32.199	82.332

P = Tweets that Passed Validation, F = Tweets that Failed Validation, T = Time Taken (MM:SS) and APS = Average amount validated Per Second.

Table 5: Validation Results (Run 1 – Using Database)

	P		F		T	APS
	#	%	#	%		
Zara	15188	12.904	102515	87.096	00:30.046	3917.427
New Look	86	0.13	66210	99.87	00:18.715	3905.263
River Island	277	3.394	7885	96.606	00:02.09	3683.962
Dorothy Perkins	103	4.396	2240	95.604	00:00.636	3542.399

P = Tweets that Passed Validation, F = Tweets that Failed Validation, T = Time Taken (MM:SS) and APS = Average amount validated Per Second.

Table 6: Validation Results (Run 2 – Excluding Database)

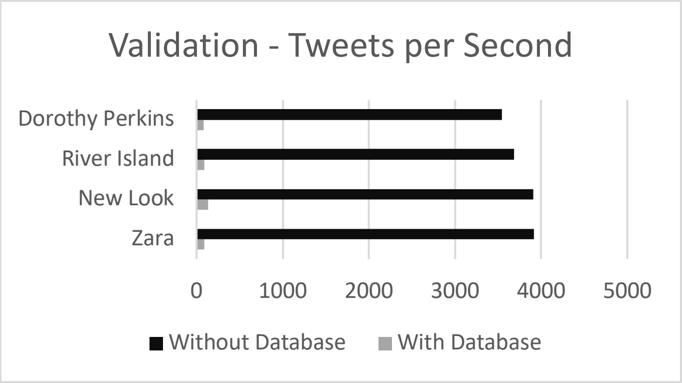

Figure 12: Tweets Validated per Second (Without Database vs. With Database)

To test accuracy of the validation process, 1000 tweets were manually sampled. Each tweet was analysed to determine if it was relevant and should pass or fail validation. These were compared to the actual results for each tweet. The test counts as passed when both actual and expected results are same. With the sample used, 946 tests passed, and 54 tests failed. This calculates to an accuracy of 94.6%. This is presented in Figure 13.

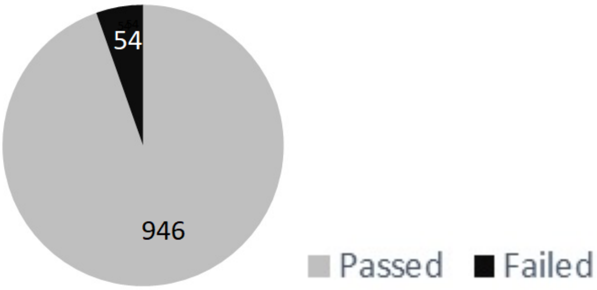

Figure 13: Validation Accuracy Test Results

Table 7 presents statistics for the Analysis module of the application, which does sentiment analysis. Like the Validation module, run 1 includes time taken to update the database whereas run 2 does not. The time taken for both runs include time to perform base sentiment analysis and sentiment scaling. As with the Validation module, a speed increase was observed with the second run, reporting completion 72 times faster on average.

	Run 1			Run 2		
	A	**T**	**APS**	**A**	**T**	**APS**
Zara	14925	01:31.4	163.293	15188	00:00.969	15673.891
New Look	80	00:00.461	173.536	86	00:00.009	9555.556
River Island	287	00:01.866	153.804	277	00:00.02	13850
Dorothy Perkins	127	00:00.912	139.254	103	00:00.015	6866.667

A = Tweets Analysed, T = Time Taken (MM:SS) and APS = Average amount analysed Per Second.

Table 7: Sentiment Analysis Statistics

When testing accuracy of calculating the base sentiment, a random sample of 1000 tweets was taken. They were then manually rated using the scoring system displayed in Table 8. 90.8% of the tweets analysed met the expected polarity and weight with 9.2% resulting in either an unexpected weight or polarity. 91.2% of the tweets analysed met the expected generalised polarity. This means that 0.4% of the tweets analysed had the correct polarity but were stronger or weaker than expected. This also results in 8.8% of analysed tweets having a different polarity than expected.

Score	Expectation
2	Positive with sentiment score of 3 or above.
1	Positive with sentiment score above 0 but below 3.
0	Neutral with sentiment score of 0.
-1	Negative with sentiment score of below 0 but above -3.
-2	Negative with sentiment score of -3 or below.

Table 8: Sentiment Scoring Table

An interesting test was to see how analysing multiple tweets within a single document would affect accuracy and speed. In the implementation, a document is classed as a single tweet. Despite increasing size of the document, providing that contained tweets were about the same product, there was no loss of accuracy when the calculated sentiment for the document was divided by the amount of contained tweets. This was tested by performing sentiment analysis on 50 tweets about the same product. On each test, the tweets were split into a varying number of equal documents (2, 5, 10, 25 and 50 tweets per document). The results, shown in Figure 14, reveals that using documents of 5 tweets was the most efficient with documents of 50 tweets being slower than individual analysis. The same tweets were used in each test, and as such the results do not include the additional time it would take to sort the tweets into appropriate documents.

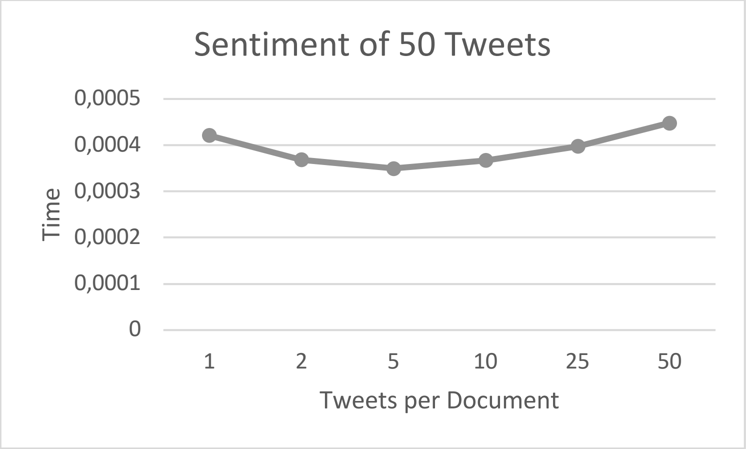

Figure 14: Speed of performing sentiment analysis on varying document sizes

Table 9 displays the number of tweets that had the required context for sentiment scaling to be performed. Zara had the highest number of tweets with the appropriate context but even this was only 12 tweets, just 0.08% of validated tweets that were analysed. On average across the two runs, 1.273% of Dorothy Perkins' validated tweets had the appropriate context. This still only equated to 2 tweets in run 1 and 1 tweet in run 2.

	Run 1				**Run 2**			
	S		**IC**		**S**		**IC**	
	#	%	#	%	#	%	#	%
Zara	12	0.08	14913	99.92	12	0.079	15176	99.921
New Look	0	0	80	100	0	0	86	100
River Island	2	0.697	285	99.303	2	0.722	275	99.278
Dorothy Perkins	2	1.575	125	98.425	1	0.971	102	99.029

S = Tweets Scaled, IC = Tweets not scaled due to Insufficient Context.

Table 9: Sentiment Scaled vs. Not Scaled

With such a low number of tweets containing enough context to be scaled, the accuracy of sentiment scaling was measured in two ways. Firstly, a random sample of 1000 tweets were manually analysed and scored using the scoring system in Table 8. This was then matched to the actual output. With the sample containing mainly tweets with no context, 99.6% of all expectations were met. An example of where the expectation was not met was in the following sentence: 'Although my valentines was last weekend, I still got a river island handbag and a soppy card. Brought a tear to my eye actually?' This was because the application misrepresented the word 'tear' (as in crying) as the damage context 'tear' (as in to rip). Due to such a low number of tweets that contained the appropriate context to be scaled, the tweets considered to have an appropriate amount in order to judge the accuracy at which the sentiment was scaled were manually analysed. Of the 15 tweets that were analysed, 14 were scaled correctly. The tweet that was not scaled correctly was: 'My cute ass jumper I got for v day has a hole in it River Island'. The word 'day' was analysed as one day ago, whereas Valentine's day was more than a day away when analysed. This gives the scaling an accuracy of 93.3%.

8 Conclusion

Consumer satisfaction forms a critical part of businesses and directly impacts on customer retention ratios. The ability to measure and define indexes for consumer satisfaction can be very useful in swiftly responding to customer needs. With consumer satisfaction data for certain products exhibiting extreme volatility because of their short requirement duration, it is necessary to identify present consumer satisfaction in a timely manner. Hence, this paper focused on measuring consumer satisfaction index in the fashion industry. Social media provides vast amounts of data for the industry, obtained via social mining solutions. However, social media data is very volatile and requires an analytic process that is suitable to address the high data volatility issue. Based on initial investigations, a software artefact with a novel sentiment analysis and scaling technique was developed, utilising social media data from Twitter for specific brands. The experimental results from the implementation validates the software regarding its benefit for use within domains with a high level of data volatility like the fast fashion industry. This has potentials to foster economic growth in the fashion industry with timely consumer sentiment analysis, and subsequently enhancement to products and services. However, challenges faced with the context of certain keywords during the analysis stage of the software suggests the availability of further research in the area.

References

[Ali, Muqadas 2015] Ali, F., Muqadas, S.: The Impact of brand equity on brand loyalty: The Mediating Role of Customer Satisfaction. Pakistan Journal of Commerce and Social Sciences (PJCSS). 2015;9(3):890-915.

[Arrigo 2016] Arrigo, E.: Fast Fashion Business Model: An Overview. In Handbook of Research on Global Fashion Management and Merchandising 2016 (pp. 186-209). IGI Global.

[Bhardwaj, Fairhurst 2010] Bhardwaj, V., Fairhurst, A.: Fast Fashion: Response to changes in the Fashion Industry. The International Review of Retail, Distribution and Consumer Research. 2010 Feb 1;20(1):165-73.

[Caro, Martinez-de-Albeniz 2015] Caro, F., Martínez-de-Albéniz, V.: Fast Fashion: Business Model Overview and Research Opportunities. In Retail Supply Chain Management 2015 (pp. 237-264). Springer, Boston, MA.

[Cerruti et al. 2016] Cerruti, C., Mena, C., Skipworth, H., Tavoletti, E.: Characterizing Agile Supply Partnerships in the Fashion Industry. International Journal of Operations & Production Management. 2016 Aug 1;36(8):923-47.

[Dhaoui et al. 2019] Dhaoui, C., Webster, C., Tan, L.: Social Media Sentiment Analysis: Lexicon versus Machine Learning. Journal of Consumer Marketing. 2017 Sep 11;34(6):480-8.

[Joshi 2019] Joshi, A.: Sentiment Analysis and Opinion Mining from Noisy Social Media Content (Doctoral dissertation, International Institute of Information Technology Hyderabad). 2019

[Lea-Greenwood 2013] Lea-Greenwood, G.: Fashion Marketing Communications. John Wiley & Sons; 2013 Mar 18. p. 31

[McNeill, Moore 2015] McNeill, L., Moore, R.: Sustainable Fashion Consumption and the Fast Fashion Conundrum: Fashionable Consumers and Attitudes to Sustainability in Clothing Choice. International Journal of Consumer Studies. 2015 May;39(3):212-22.

[Madhani 2015] Madhani, P.: Enhancing Customer Lifetime Value in Fast Fashion Retailing with RFID Initiatives. International Journal of Business and Globalisation. 2015;15(2):205-37.

[Muhammad et al. 2016] Muhammad, A., Wiratunga, N., Lothian, R.: Contextual Sentiment Analysis for Social Media Genres. Knowledge-Based Systems. 2016 Sep 15;108:92-101.

[Oliver 2014] Oliver, R.: Satisfaction: A Behavioral Perspective on the Consumer: A Behavioral Perspective on the Consumer. Routledge; 2014 Dec 18.

[Paltoglou, Thelwall 2012] Paltoglou, G., Thelwall, M.: Twitter, MySpace, Digg: Unsupervised Sentiment Analysis in Social Media. ACM Transactions on Intelligent Systems and Technology (TIST). 2012 Sep 1;3(4):66.

[Paterson 2017] Paterson, M.: Consumption and Everyday Life. Routledge; 2017 Oct 25.

[Sreekumar, Rajmohan 2019] Sreekumar, V., Rajmohan, M.: Supply Chain Strategy Decisions for Sustainable Development using an Integrated Multi-Criteria Decision-Making Approach. Sustainable Development. 2019 Jan;27(1):50-60.

[The Washington Post 2019] The Washington Post. (2019). Twitter reveals its daily active user numbers for the first time. [online] Available at: https://www.washingtonpost.com/technology/2019/02/07/twitter-reveals-its-daily-active-user-numbers-first-time/?noredirect=on [Accessed 8 Aug. 2019].

[Turker, Altuntas 2014] Turker, D., Altuntas, C.: Sustainable Supply Chain Management in the Fast Fashion Industry: An Analysis of Corporate Reports. European Management Journal. 2014 Oct 1;32(5):837-49.