An Automated Cloud-based Big Data Analytics Platform for Customer Insights

Liangxiu Han[1*], Muhammad Salman Haleem[1], Tam Sobeih[1], Ying Liu[2], Anthony Soroka[3] and Lianghao Han[4]
[1]School of Computing, Mathematics & Digital Technology, Manchester Metropolitan University, Manchester, M1 5GD, UK
[2]Cardiff School of Engineering, Cardiff University, The Parade, Cardiff CF24 3AA UK
[3]Cardiff Business School, Cardiff University, Colum Drive, Cardiff CF10 3EU UK
[4]School of Medicine, Tongji University, 1239 Siping Road, Shanghai, P.R.China
Email: l.han@mmu.ac.uk

Abstract— **Product reviews have a significant influence on strategic decisions for both businesses and customers on what to produce or buy. However, with the availability of large amounts of online information, manual analysis of reviews is costly and time consuming, as well as being subjective and prone to error. In this work, we present an automated scalable cloud-based system to harness big customer reviews on products for gaining customer insights through data pipeline from data acquisition, analysis to visualisation in an efficient way. The experimental evaluation has shown that the proposed system achieves good performance in terms of accuracy and computing time.**

Keywords- automated sentiment analysis; cloud computing; product reviews, business intelligence

I. INTRODUCTION

As the arrival of the fourth industrial revolution, data-driven digital services and products are playing a central role to drive revenue growth, optimise customer interaction, and offer strategic decisions for both companies and customers on what to produce or buy [1][2].

One area of interest is in extracting and quantifying customer's affective states and other subjective information from online reviews appearing on social media, as well as merchant and review websites, for various products and services. A product review written by a user, refers to a customer's opinion about a product, which describes negative, positive or neutral parts of a product. Product reviews have been widely recognised to have a significant influence on customers' shopping decisions, products and services, business strategies and manufacturing decisions [1][3][4][5]. For instance, Ghose and Ipeirotis [4] presented a comprehensive analysis on the impact on economic outcomes, such as product sales and its relation to product reviews, by exploring multiple aspects of review text. In [5], the authors described how customers read reviews to find unique information about products and consequently reduce the risk of their buying decision. Archak and Ipeirotis [2] also developed an approach to mining consumer reviews, in order to better understand what are the customer preferences and actions, how reviews impact the price power of products and how to improve sales forecasts.

However, with the massive information available, manual analysis of the data is costly and time consuming, as well as being subjective and prone to error. For example, if a customer wants to buy a product, he or she might only be able to read a small number of reviews, which may lead to a biased purchase decision. Similarly, a business may not be able to keep track of their products and understand market needs and make timely decisions on what to produce. Despite encouraging research efforts on sentiment analysis in the domain of product reviews [1][6][7], most existing systems are partially automated with the main focus on sentiment analysis part, which may not be able to handle big data. In addition, these systems mainly provide text review summaries rather than user-friendly graphical summaries. The challenges remain on how to collect data from websites, analyse them and present the results to users in a user-friendly and timely manner.

To address the challenges above, different from all existing systems, we have proposed a cloud-based data analytics platform, which can automatically scrape data from online websites, perform the sentiment analysis of a given product and visualise the summaries of reviews based on product feature aspects and aggregated features, in order to gain customer insights in real time.

The rest of this paper is organised as follows: Section II describes the proposed system and methodology; Section III presents the experimental evaluation; Section IV concludes the work and highlights future work.

II. RESEARCH DESIGN AND METHODOLOGY

A. *Aims and Objectives*

The goal of this work is to fully automate the process of big customer review data and develop a scalable, user-friendly system to gain customer insights and inform strategic decisions in a timely manner. Specifically, the main task is to collect data from online websites and categorise the review texts into one specific sentiment polarity, positive or negative (or neutral), and provide product summarisation based on features of a given product in real time. Essentially, it logically forms a data ingest pipeline from data collection, data analysis, interpretation and visualisation. The objectives of the system design include:

1) To facilitate a user search on a product and its review data.

2) To acquire the review data and perform a sentiment analysis on it to extract useful features and their sentiments.

3) To interpret and visualise the results of the analysis in a meaningful way to highlight the effectiveness of the analysis.

4) To provide an efficient and easy-to-use web-based interface to control the system.

B. *The Proposed System Architecture and Methodology*

Based on the objectives above, the system should be able to provide the ability for a user to search for products, select them and view a number of charts that highlight the customers' affections to various features of the product. Useful information that can be gleaned includes, but is not limited to, overall satisfaction with the product, a list of features specifically chosen for comment by customers and their related sentiments, the ability to compare categories for a given polarity to see the distribution of sentiment, the distribution of sentiment for any given category and the comparison of these charts for different products. The system should also be able to show the potential for geographical maps if the data is available, highlighting countries with the most liked or disliked products or areas with the largest number of reviewers or particular sentiments. Finally all of these can be linked directly to the text of the review, allowing the user to select any sections of the charts and read the filtered reviews specifically related. This allows the user to read why there is a negative sentiment regarding price or what exactly people like about the way it fits. All this information can be used to better improve products or services by detecting trouble areas or showing what are working well.

The overview of the system is shown in Fig. 1, consisting of several components: 1) Data acquisition; 2) Sentiment analysis; 3) Data visualization; 4) Graphical User interface.

1) Data acquisition

To analyse the reviews, it is important to accurately collect data from websites in a timely manner first. Due to high volumes of data, manual collection of web data increases the cost of labour significantly. Additionally it is known to be error-prone. Therefore, there is a need to develop an automated approach to scraping review data from websites. In this work, we have developed a web scraping service to collect review data from websites, which can be effortlessly consumed by analysis components of the system.

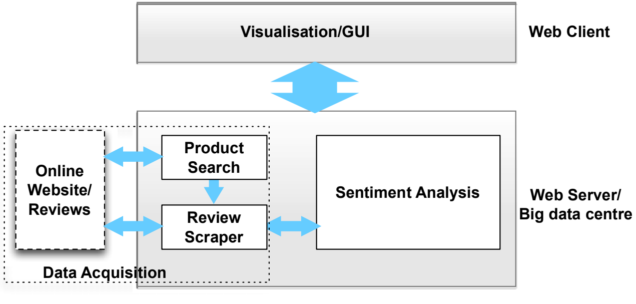


Fig. 1. System overview

2) Sentiment analysis

Sentiment analysis refers to detect whether a given text represents a positive or negative or neutral opinion (sentiment polarity). There are three levels of sentiment polarity classification, including document, sentence and aspect levels [8]. The document level classification mainly focuses on expressing a positive or negative opinion on a document. The sentence level analyses each sentence' sentiment polarity. The aspect-level sentiment analysis mainly deals with specific aspects or features of a document. In our case, we mainly target aspect-level sentiment analysis of product features. For example, a review about a mobile phone may contain the following description:

"**The screen was great but the speakers were terrible**". An ideal sentiment analysis system should be able to identify both "screen" and "speakers" as product features with positive and negative sentiments, respectively.

Aspect-level sentiment analysis is particularly useful for comparing two or more products based on their features. However, most of current online websites only provide an overall score for a particular product. Consumers have to read through all reviews and conduct manual analyses to make decisions.

In our work, we have applied both document-level analysis and aspect-level analysis to product reviews. The workflow of sentiment analysis module is shown in Fig. 2.

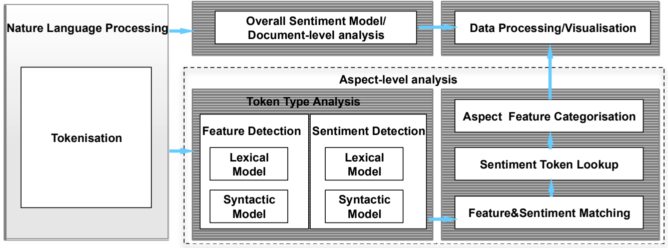


Fig. 2. Sentiment analysis workflow

As shown in Fig.2, the Natural Language Processing module (NLP) takes a review text as input and pre-processes the data by performing tokenisation on the texts for further aspect-level and document level analyses. The aspect analysis module takes the output from the NLP module, identifies features and sentiments, and performs aspect feature categorisation, which can then be visualised through a graphical user interface. Similarly, the document level analysis performs sentiment analysis of the whole review text, whose outputs will be displayed to the user.

a) Aspect-level sentiment analysis

For the aspect-level sentiment analysis of this work, we have built upon the work [6] and extended it by introducing a new feature categorisation and a feature matching algorithm. The basic technologies in use are the Stanford Natural Language Processing library (coreNLP) for tokenisation [9][10] and Support Vector Machine (SVM) using the libSVM library implementation for feature detection and sentiment detection [11]. Essentially, it attempts to determine if any given word in the review is a 'feature' or a sentiment-bearing word, through first processing the review text using coreNLP which tokenises the text and assigns features to each token based on natural language principles, and then passing them into four SVM models (both lexical and syntactic models for 'feature' and 'sentiment' detection separately). These models provide a score based on how much each token fits each role and if a token has scores above zero, and the model providing the highest score determines its type. Once the features and sentiments have been extracted, it determines which sentiment applies to which feature through the use of the matching algorithm developed for this project. Then, all the raw review texts packaged up with their specific extracted features and sentiments are sent to the web application.

During this process, it is critical to match each feature with the appropriate sentiment bearing word so that the correct sentiment value can be recorded. We have developed a rule-based matching algorithm to connect a feature with its appropriate sentiment working on a proximity basis and then with any connected sentiment found through the natural language processing. This method consists of three steps:

- The first step checks whether the feature itself is sentiment bearing, if so, matches the token with itself since the sentiment expressed by the token certainly is related to the feature, i.e. "comfortable".
- The second step iterates through the surrounding 6 tokens, 3 on each side of the feature looking for a sentiment. If a sentiment is found then it is matched, because a sentiment closer to a feature is more likely to be related to the feature. For example, many of the reviews have simple terms such as, 'good looking' or 'great price', or 'the colour was good' or 'it fits very well' etc., the features and sentiments can be related through checking the 6 tokens alternately, starting with the preceding one and then the next one and continuing towards any detected sentiment with a high probability of being connected. However a number of issues may arise from this particularly lists and comments including 'but'. For instance, 'looks good but price was terrible', in this case the algorithm would match 'price' with 'good' at this stage even though both sentiment words are equidistant from the feature. So whenever a 'but' is found, the algorithm is stopped from checking any more in that direction and the 'but' clearly denotes a separate comment.
- The final step checks the semantic incoming edge of the feature token (which is set as part of the natural language processing module) to determine if it is a sentiment or a feature. If it is a sentiment, then with the previous reasoning it is likely to be connected to the feature as it is in semantic proximity. If it is a feature and there is no sentiment within a 3 word radius, we can assign the sentiment of the new feature to the current one as it is likely with a list or comment discussing the same sentiment about multiple features at the same time, given the fact that the feature is semantically close. To do so, the matching algorithm is recursively called on the new feature and the result is returned. Finally the other semantically tagged tokens are inspected.

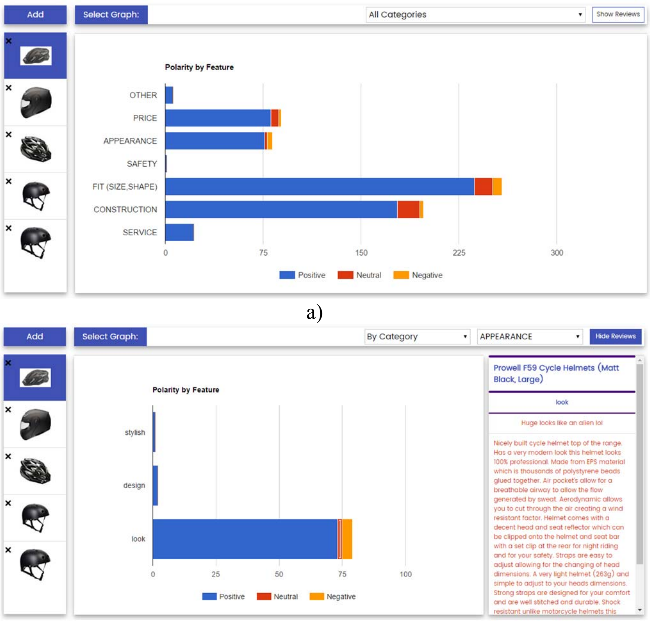
b) *Document-level sentiment analysis*

For the document-level analysis, we have analysed the sentiment polarity of a whole review text based on the Stanford Natural Language Processing library (coreNLP) [9][10], which takes the input of a review text and returns the polarity of each sentence and then aggregates the polarity of sentences.

3) *Data visualization*

To provide a clearly condensed representation of the reviews about a given product that a user wants to get, we have developed visualization techniques in various types of charts to provide the summaries of product reviews based on features and or aggregated features of a given product. The charts have been chosen to provide insights based upon the results of the analysis, but also to show that the data can be displayed in various ways. These charts include:

- A stacked bar chart illustrating positive, neutral and negative reviews for all features, as shown in Fig.3 a) and b). Fig.3 shows how customers feel about specific aspects of a product. It is easily to see that most customers care about "Fit" and "Construction" with the majority positive reviews in Fig.3a.
- A pie chart presenting multiple products with positive, neutral and negative for each feature, as shown in Fig.4. This chart allows users to quickly identify which products, categories and features have strong positive or negative opinions.
- A column chart in Fig. 5. presenting multiple products with positive, neutral and negative for each feature to enable a comparison among the product features.
- A geo-location chart in Fig.6. illustrating positive, neutral and negative reviews by countries of manufacturers.



a)

Prowell F59 Cycle Helmets (Matt Black, Large)

look

Huge looks like an alien lol

Nicely built cycle helmet top of the range. Has a very modern look this helmet looks 100% professional. Made from EPS material which is thousands of polystyrene beads glued together. Air pocket's allow for a breathable airway to allow the flow generated by sweat. Aerodynamic allows you to cut through the air creating a wind resistant factor. Helmet comes with a decent head and seat reflector which can be clipped onto the helmet and seat bar with a set clip at the rear for night riding and for your safety. Straps are easy to adjust allowing for the changing of head dimensions. A very light helmet (263g) and simple to adjust to your heads dimensions. Strong straps are designed for your comfort and are well stitched and durable. Shock resistant unlike motorcycle helmets this

b)

Fig. 3. A stacked chart to provide summarisation of polarity for each category of a given product: polarity by category without showing review texts a) and b) with review texts.

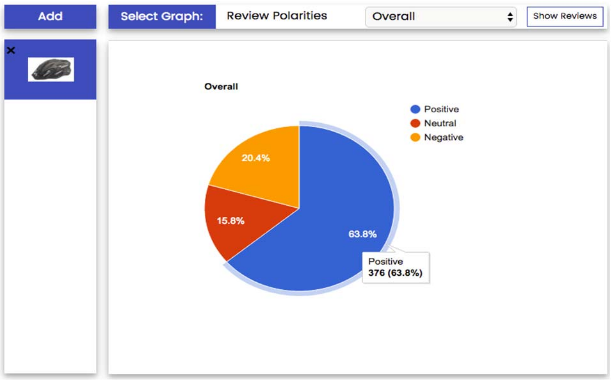


Fig. 4. A pie chart to sumarise the polarity of reviews

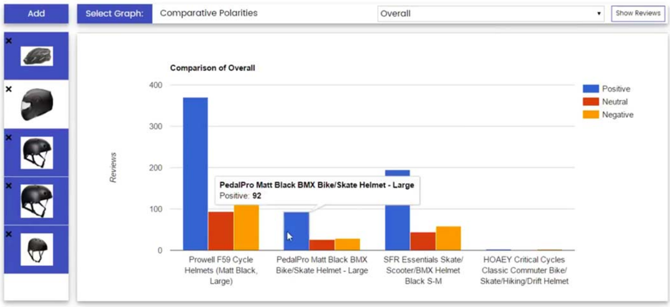

Add
Select Graph:
Comparative Polarities
Overall
Show Reviews
Comparison of Overall
400
300
200
100
0
Reviews
Positive
Neutral
Negative
PedalPro Matt Black BMX Bike/Skate Helmet - Large
Positive: 92
Prowell F59 Cycle Helmets (Matt Black, Large)
PedalPro Matt Black BMX Bike/Skate Helmet - Large
SFR Essentials Skate/ Scooter/BMX Helmet Black S-M
HOAEY Critical Cycles Classic Commuter Bike/ Skate/Hiking/Drift Helmet

Fig. 5. A column chart to show multiple product features for easy comparison

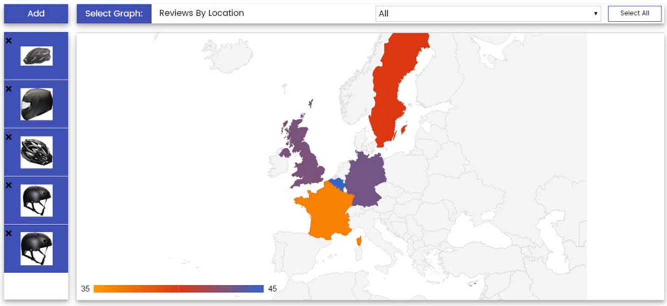


Fig. 6. A geo chart to show positive, neutral and negative reviews by countries of manufacturers (The colour of the country represents a normalised polrity)

4) The graphic user interface

One of the objectives of this work is to design a user-friendly easy-to-use web-based interface to control the system. With this in mind, we have used different cards to delineate a number of functional areas, which contain a different aspect of the interface. We have also used a small palette of contrasting colours to clearly emphasise the function of items, such as buttons and dropdown menus. Shadows are also used to simulate the depth on interactive objects. The changes of colour and shade are used to denote the state of user controls coupled with the changes to the pointer standardised across browsers. The chart card is used to display the visualised data of a selected item that can be controlled with the contextual controls in the card above. This card contains a chart selection dropdown to change between different charts and a range of select inputs and buttons that will appear based on the chosen chart to specify the data being displayed.

C. The Implementation

To enable the system to execute on the cloud computing platform and ensure the scalability and the efficiency, we have chosen a REST design pattern, all the system components have been implemented as web services based on Dropwizard [12], a java-based framework/library to facilitate the development of high performance, restful web services. The system combines Jetty as an embedded web server, with Jersey to map Java objects directly to HTTP requests using Jackson for the JSON conversion and a variety of commonly used libraries in Java development. Essentially it allows us to create standard Java classes and use annotations to map various methods and objects to the HTTP request URLs without having to manually create handlers or servlets, etc. It also generates a single large jar file which is combined with a configuration file to run the service, making the distribution and the deployment in a simple matter.

III. EXPERIMENTAL EVALUATION

To evaluate our proposed system, we have conducted a use case study by investigating product reviews of "Helmet" to understand how the system can harness big data to gain customer insights.

A. Data Description

For the experimental purpose, as a pilot study, the data was obtained from Amazon website. Since we have used a supervised machine learning approach, for the training purpose, we have first collected a total of 704 reviews containing 1073 product aspect features for the model construction. Each feature was tagged with the exact text from the review. N-fold cross validation was used for the validation of the model.

B. Evaluation Metrics

We have evaluated the proposed system in terms of accuracy and computing time from data collection to visualisation.

1) Accuracy evaluation

Precision, recall and accuracy have been used to measure the accuracy performance of analysis.

Precision refers to the percentage of all positive predictions that are correct.

***Precision* = True positive / (True positive + False positive)**

Recall refers to the percentage of all positive samples that are correctly predicted.

***Recall* = True positive / (True positive + False negative)**

Accuracy refers to the percentage of all predictions that are correct.

***Accuracy* = (True positive + True negative)/(True negative + true positive + false negative + false positive)**

The evaluation results of both document-level and aspect-level analyses are listed in Table 1. The experimental results show that the feature aspect-level sentiment analysis with our matching algorithm has a high accuracy, compared to the result using the method proposed in [6]. The work in [6] doesn't have a document-level sentiment analysis. Therefore, there is no comparison result available for comparison. We only list our document-level sentiment analysis in Table 1.

TABLE I. ACCURACY EVALUATION

Type	Evaluation Metrics		
	Precision	*Recall*	*Accuracy*
Feature aspect-level sentiment analysis using our own matching algorithm	0.89	0.97	0.87
Feature aspect-level sentiment analysis using the method in [11]	0.38	0.86	0.37
Document-level sentiment analysis	0.68	0.43	0.53

C. Processing time and scalabiliyy

To evaluate the processing speed and the scalability of the proposed system, we have tested it on a number of reviews/words and calculated the computing time for each main component, including scraping, tokenisation and analysis time under conditions of different number of reviews and words and different CPUs.

1) Total execution time – number of reviews/words - CPUs

As shown in Fig.7 and 8, the execution time increases with the increasing number of reviews. In addition, the execution time reduces with the increasing number of CPUs. The result shows an approximately linear relationship between the number of reviews and the total execution time. The regression equations for the number of reviews (under different numbers of CPUs) are shown as follows:

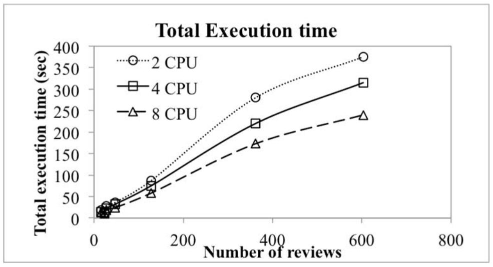


Fig.7. The relationship between total excution time and the number of reviews with different CPUs

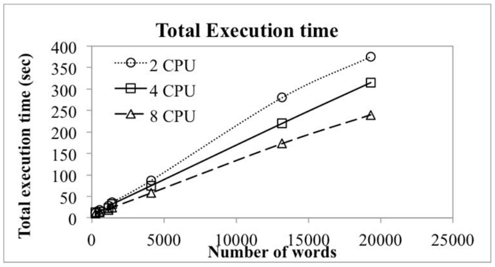


Fig.8. The relationship between total excution time and the number of words with different CPUs

$$y = 0.0046x + 4.6661 \quad R^2 = 0.927, \quad 2CPU$$

$$Y = 0.5294x + 7.7109 \quad R^2 = 0.993, \quad 4CPU$$

$$Y = 0.4052x + 6.8486 \quad R^2 = 0.989, \quad 8CPU$$

The regression equations for the number of words (with different CPUs) are shown as follows:

$$y = 0.0195x + 7.752 \quad R^2 = 0.997, \quad 2CPU$$

$$Y = 0.016x + 7.507 \quad R^2 = 0.999, \quad 4CPU$$

$$Y = 0.0123x + 6.5816 \quad R^2 = 0.999, \quad 8CPU$$

The R-squared values are high under different numbers of CPUs, showing a strong linear relationship.

2) The computing time for each component of the system with differen number of CPUs

We have also compared the computing time of each main component of the system including scraping time, tokenization and analysis time for different products (with different numbers of reviews and words), as shown Fig.9. In Fig.9, time to scrape has a strong linear relationship with the number of reviews but has no linear correlation with the number of words. As the number of reviews and words increase, the total time to analysis increases. The time to tokenisation has no linear correlation with the number of words or reviews. The time to analysis has an approximate linear relationship with the number of reviews or words. In all cases, with the number of CPUs increasing, the computing performance is improved.

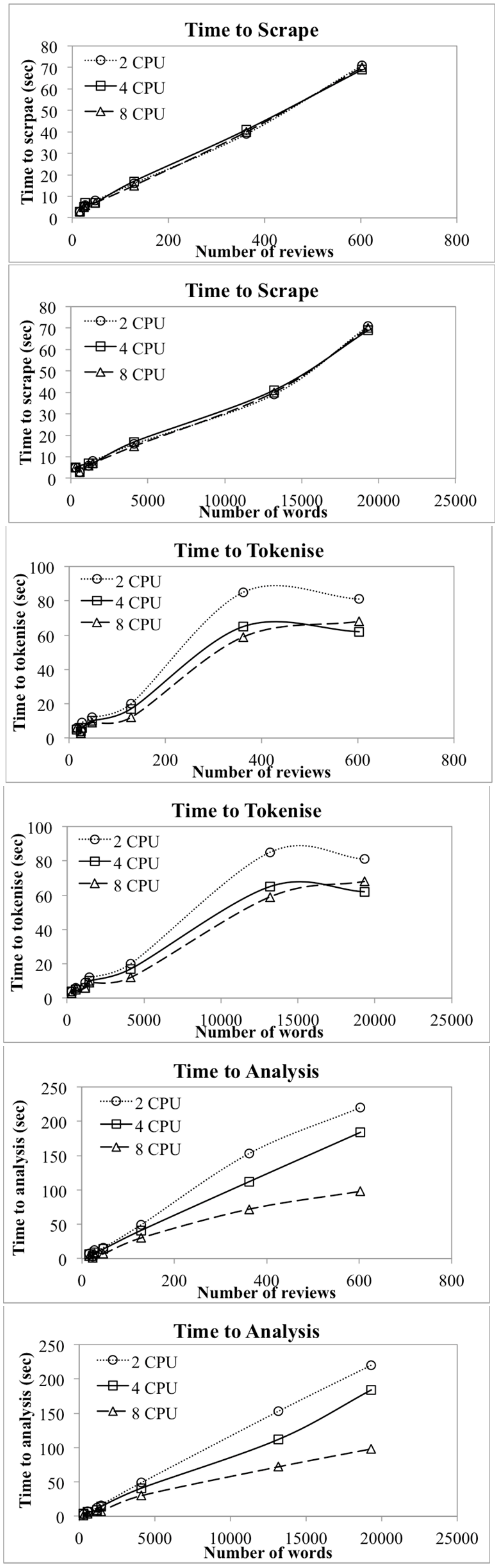


Fig.9. The relationship between analysis time of each main component of the system, the number of reviews, words and the number of CPUs

IV. CONCLUSION

Customers are at the centre of the changes to value chains, products and services. Using smart data analytics is an efficient way to understand and meet their needs. This paper has proposed a scalable, user-friendly system to enable fast automated extraction of product features and classification of polarity of product reviews to gain customer insights. The experimental evaluation shows that the proposed system has a high accuracy on feature aspect-level sentiment analysis. The relationship between the number of reviews/words and the total execution time has been investigated under different CPUs. A strong correlation with a linear relationship has been observed between them, which demonstrates the scalability of the system.

ACKNOWLEDGMENT

This work forms part of the EPSRC RECODE Consumer Goods, Big Data and Re-Distributed Manufacturing (EP/M017567/1) Redistributed Manufacturing Network. Conducted through the SME-Oriented Collaborative Customer Insight Platform for User-driven RdM (SOC-UDRdM) project.

REFERENCES

[1] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, “Semeval-2015 task 12: Aspect based sentiment analysis,” in the 9th International Workshop on Semantic Evaluation (SemEval 2015), 2015.

[2] N. Archak and P. G. Ipeirotis, “Show me the money! deriving the pricing power of product features by mining consumer reviews,” in KDD ’07 Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 56–65, 2007.

[3] A. Duric and F. Song, “Feature selection for sentiment analysis based on content and syntax models,” Decision Support Systems, vol. 53, no. 4, pp. 704–711, 2012.

[4] A. Ghose and P. G. Ipeirotis, “Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics,” IEEE Transactions on Knowledge and Data Engineering, vol. 23, no. 10, pp. 1498–1512, 2011.

[5] M. Burton, Jamie; Khammash, “Why do people read reviews posted on consumer-opinion portals,” Journal of Marketing Management, vol. 26, no. 3, pp. 230– 255, 2010.

[6] D. H. Carter, “Inferring asepect-specific opinion structure in product reviews,” Master’s thesis, School of Electrical Engineering and Computer Science, 2015.

[7] X. Fang and J. Zhan, “Sentiment analysis using product review data,” Journal of Big Data, vol. 2, no. 1, pp. 1–14, 2015.

[8] B. Liu, sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies. Morgan and Claypool Publishers, 2012.

[9] Stanford CoreNLP -- a suite of core NLP tools, http://stanfordnlp.github.io/CoreNLP/, Retrieved on 12, March 2017.

[10] C. D. Manning, J. B. Mihai Surdeanu, J. Finkel, S. J. Bethard, and D. McClosky., “The stanford corenlp natural language processing toolkit,” in the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55–60, 2014.

[11] C.-C. Chang and C.-J. Lin, “LIBSVM -A Library for Support Vector Machines”, retrieved on 12, March 2017.

[12] DROPWIZARD, http://www.dropwizard.io/1.0.6/docs/