SOCIAL MEDIA SENTIMENT ANALYSIS FOR BRAND MONITORING

NWOHIRI, A.M.[1]* AND AMAECHI, C.G.[2]

[1]Department of Computer Sciences, University of Lagos, Lagos, Nigeria.
[2]Eden AI, Johannesburg, Gauteng, South Africa.

ABSTRACT

Social media (SM) has had a profound effect on the business ecosystem. It provides immense potential for businesses because consumers habitually log on to it daily and are exposed to companies. Companies depend on sentiment analysis (SA) to gain a deeper understanding of the consumer mindset. SA can be of assistance so that you gain insights about new markets, foresee industry trends, and most importantly, understand what didn't go well with previous product releases to help you improve your existing and future products and services. Obtaining an accurate and truly useful information from SM sentiment analysis has become a challenging issue in recent years. In this paper, we propose a web-based brand monitoring system that helps businesses and organizations monitor public sentiments or opinions about their brand, product or services. SM platform Twitter was used as a case study, involving the 2016 United States presidential election twitter dataset. The study utilized statistics, natural language processing, and machine learning to determine the emotional meaning of communications. The developed system facilitates informed decisions and resolution of public complaints efficiently without the need for crude analysis (customer surveys, product forms, etc).

Keywords: Crude analysis, natural language processing, opinion mining, sentiment analysis, social listening, twitter

***Correspondence:** anwohiri@unilag.edu.ng

INTRODUCTION

SM is a collective term for websites and applications that focus on communication, community-based input, interaction, content-sharing and collaboration. Some popular examples of general SM platforms include Twitter, Facebook, LinkedIn, Instagram, and TikTok. The amount of data generated on SM has become so large that researchers and analysts are beginning to use it as resources to gain in-depth knowledge and insights across many fields of interest.

Amid the number of SM media available on the Internet, Twitter has become one of the most popular and active. It provides the largest source of user-generated data because it is a highly sociable and authentic channel [1]. The way consumers search for information about products and services has significantly changed over time [2]. Offline methods such as newspaper or magazines have been replaced by e-marketing, such as SM websites for online reviews of services and comments [3]. In recent times, consumers now use SM sites as a means to consult about brand products or services before making decisive purchases, especially other consumers' attitudes and experiences regarding a product/service. About 48% of shoppers would check a brands SM presence before confirming a purchase [4]. The numbers are staggering, 74% of consumers rely on SM to guide purchasing decisions. Before they'll add a product to their baskets, they check peer reviews and comments. A further 60% of consumers interact on SM with the brands they buy from [5].

Achieving positive brand sentiment is an imperative for all companies because it leads to more customer loyalty and better business results. Brands must make sure that negative sentiments are duly attended [6]. Increasing customer interest improves the company's image and brand loyalty, as well as influencing buying decisions. Consumers tend to trust companies or brands with a positive consumer review or reputation and are even likely to pay more for their products and services. In a study by Hudson *et al.* [7], the authors emphasized the importance of emotions in the process of forming consumer-brand relationships, as well as their effect on consumers' willingness to recommend a product. NLP techniques are used to determine whether consumer's sentiments and reviews are positive, negative or neutral [8]. NLP has been a trending topic in recent times and will be used in this study in building an efficient brand monitoring system.

Since consumers today respond more favourably to content shared on SM than to traditional advertising campaigns [9]. The emotional reaction to the first strengthens the brand's relationship with customers. When customers can engage with brands, the favourable online presence is achieved by promoting the business on SM, brand awareness, customer feedback, or recommendations. Customer expectations lead to a positive or negative emotional reaction, as a result, their happiness is shaped by negative feelings, loyalty and trustworthiness. It is relevant for brands to perform adequate analysis on online user-generated content and especially analysing the hidden emotions and sentiments in the content. This will help drive better information in brands improving their products and services. This solution must be cost effective in comparison to already existing tools that are costly for brands in developing countries.

Considering the problems above, there is a need for a brand monitoring tool to be developed. There are already several existing brand monitoring tools, some of which even tackle some of the problems above, but they all have their shortcomings. This paper also aims to enlighten researchers and developers to the shortcomings and to the possible solutions suggested within. We also compare our tool with already existing solutions. Businesses are willing to go to any extent to improve their quality of service. Such a monitoring system would provide better analysis into customers' opinions, helping businesses to make informed decisions. It is centered on helping businesses in developing countries make better, adequate decisions and ensure that customers are happy with the products and services delivered. This in turn will boost the economy of the developing countries. This study places emphasis on developing countries due to the high cost of existing brand monitoring tools.

MATERIALS AND METHODS

Brand monitoring is beyond no doubt being appreciated by a diverse community of brands, consumers and organizations that have encouraged the technology thus far. However, the existing technologies are far too expensive to be purchased by brands in developing countries. This feature is crucial to brands or organizations that may want to improve the efficiency of products or services by leveraging on analysing public sentiments. This work proposes a design concept that should be able to manage this challenge if adopted. The new system is a web-based system application that allows brands to monitor public sentiments through SM using Twitter as a case study. It allows brands or organizations, aided by charts and graphs of consumer sentiments and emotions, to make better decisions. The system allows brands to view the positive, neutral and negative tweets in the dashboard. Our proposed users would like to view their dashboard and collate different analytical formats for business analysis and informed decisions.

In order to make the brand monitoring system experience as successful as possible, easy to use and efficiently developed, more time is dedicated to overall systems design, outlining requirements and modelling the business objects and the relationships that exist using entity relationship, context data flow, and architectural diagrams. The entire system is based on the model–view–controller (MVC) architectural pattern [10, 11, 12]. However, the following tools and technology and practices are put into use: Visual Studio text code editor is used for writing and debugging the web app; Figma templating engine is used for the systems user interface (UI) design; Cascading style sheets version 3 (CSS3) is used to achieve design responsiveness; Node.js is used for core programming backend and JSON is used for information transfer and application configuration; React js is used as a front-end framework and would help display the needed charts and graphs for analysis.

Requirements specification

The system requirement needed in the development of this project is divided into two parts: functional and non-functional requirements. The functional requirement defines a function of a system or its components. Functional prerequisites might be computations, specialized subtleties, information control and preparing, and other explicit usefulness that characterize what a system should achieve. Listed below are some of the functional requirements. The system fetches data associated with the token from the 2016 U.S. presidential election twitter dataset; It extracts necessary features from fetched twitter dataset; It predicts the sentiments of the tweets the associated token is mentioned or tagged; It displays analysis results using charts and graphs. The non-functional requirements refer to the criteria that can be used to measure the performance of the overall system. Some of the non-functional requirements include: speed, availability, flexibility, interface, security and usability (see Table 1).

Table 1: Non-functional requirements

Requirements	Description
Speed	There should be no delay in responding to a request
Availability	The system must be readily available at anytime
Flexibility	The system must be able to adapt to possible updates in the future
Interface	The application's interface must be user-friendly
Security	Analysed data and results are not saved in the system; hence users are assured of data security
Usability	The system will be easy to use especially for a novice.

System design

The system design presents a simple way of categorizing the modules and relationship between entities. Figure 1 below presents our context diagram, which shows the interactions between the system and other actors with which the system is designed to interface. This refers to the overview structure of the system. The context diagram presented in Figure 1 shows the various relationships and interactions that go on in the system. This refers to the overview structure of the system.

We adopted the Client-Server architecture. A system and a server database establish a connection using TCP/IP protocol. Data processing is done in the data layer. The system is optimized for both web application and mobile interactions. The data layer consists of the authentication middleware which guarantees that only authorized users are allowed to use the system, the model engine runs analysis on the user twitter data, the visualization engine accepts the post-processed data and sends a notification to the user. The data is updated in the user schema and then communicated to the client. Figure 2 shows the proposed architecture.

The ideas of use case and action graphs are additionally used to clarify the working of the system. Figures 3a and 3b below represent the flow chart and use-case scenarios, showing the system actors and their respective roles (use-cases). After logging in, the roles of each user are as follows: User – registers with email and analyses dashboard to make better decisions and receives notifications on negative sentiments.

The use-case describes what actions need to take place before the actors can perform their roles, it also depicts the state of activities by showing the sequence of activities performed by possible workflow behaviour of the system. When the twitter tokens are inputted, the related tweets or dataset is fetched from the 2016 U.S. presidential election twitter dataset and pre-processed by removing stop words, stemming words and mapping it to the root word. The dataset is matched with a trained dataset and relevant values are obtained. All necessary information or analysis results will be outputted and displayed to the user.

The Entity Relationship diagram (ERD), shown in Figure 4, is a graphical depiction of an information structure that depicts the associations among people, objects, spots, thoughts or events inside that system. An ERD is a data exhibiting strategy that can help with describing business gauges and be used as the foundation for a social information sharing index.

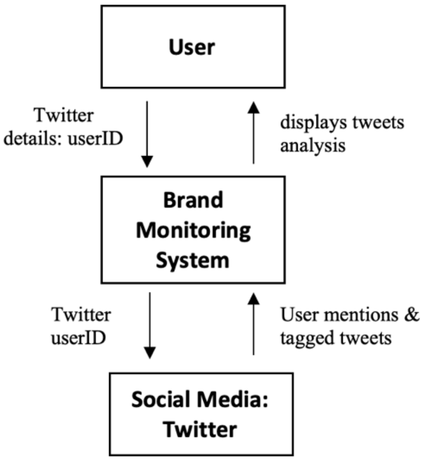


Figure 1: Context diagram of the proposed system

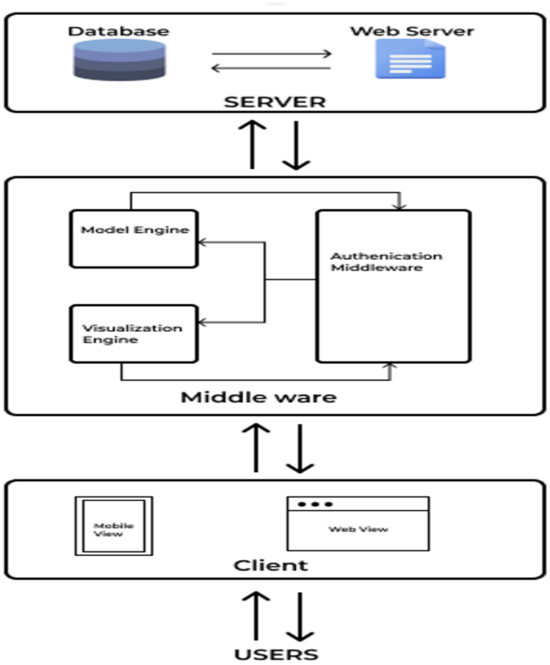


Figure 2: Client-Server architecture of the proposed system

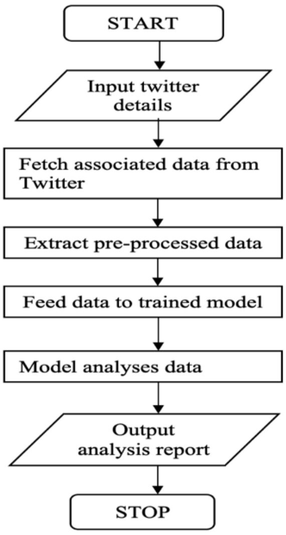

START
Input twitter details
Fetch associated data from Twitter
Extract pre-processed data
Feed data to trained model
Model analyses data
Output analysis report
STOP

Figure 3a: Flow Chart of the Proposed System

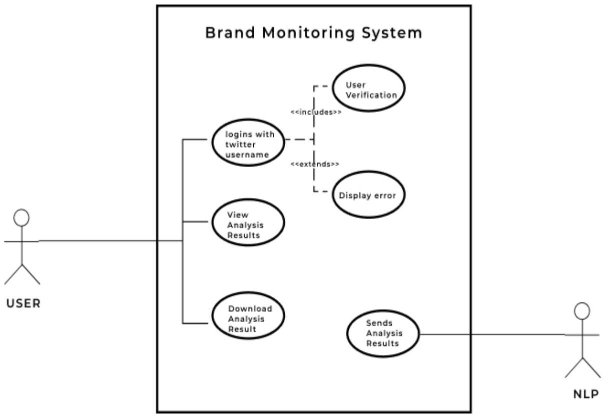


Figure 3b: Use case diagram of the Proposed System

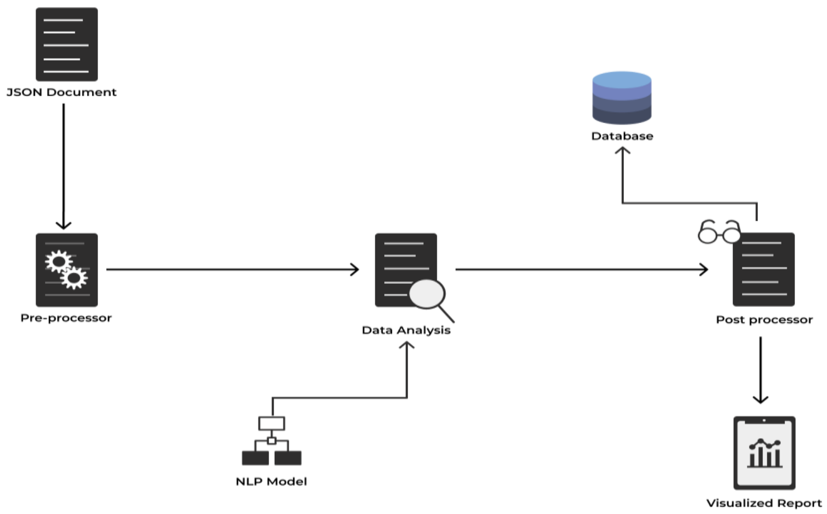


Figure 4: E-R Diagram of the Proposed System

Components of the ER Diagram are explained in detail in Table 2.

Table 2: Components of the Entity Relationship Diagram

Component	Description
JSON Document	This is the object format that contains the user details. It stores the tweet or mention, or the tag associated with the user. The tweets associated with the token are passed in to be analysed and will also be stored in a dataset to be used as a reference to analyse other tweets.
Pre-processor	This is an entity that acts on the tweets. It performs data cleaning and ensures it is ready to be acted upon. During this stage, all stop words are removed. This allows for words to be better matched, acted upon accurately and removes all unnecessary words that would have been stored otherwise.
NLP model	An NLP Model would be generated for Naïve Bayes such that tweets which are conceptually similar to each other would have values close to each other. The sentiments of each tweet would be classified into its similar or close matches from the trained datasets.
Post processor	In this stage, the information gathered from the previous entities is stored in the database for reference purposes. It is also sent to the visualized report to be displayed to the user.
Visualized Report	This is the output from the brand monitoring system process. The information is carefully displayed in bars, charts and graphs for the user to conduct analyses, respond to negative sentiments and make important business decisions.

Developmental tools

The tools which include but are not limited to Software Development Kits (SDKs), frameworks, programming languages and Application Programming Interfaces (APIs), Jupyter Notebook, Database Management Systems (DBMS) that were combined in the implementation of this work are highlighted and discussed briefly in Table 3.

Table 3: Developmental tools

Tools and Libraries		**Technology**	
Visual Code	The editor used to write the code	**React js**	This enables the app to render the user interface to the user, as well as communicate to the server via JavaScript.
Git	This was used for version control	**Node**	A server-side environment, written with JavaScript. It was used to handle rendering of resources to the front-end, database connection and management, as well as other back-end operations.
Tailwind CSS	The CSS library used for the user interface	**MongoDB**	A NoSQL database, or document database. It was the database technology employed for this app
jQuery	The front-end library used for DOM handling		

IMPLEMENTATION & EVALUATION

System implementation

Here we evaluate the effectiveness of the methods earlier described. We gave our system the name *Pisocial*. The implementation stage involves the code-and-fix method of development. The programming language used for the implementation of this project is JavaScript. JavaScript is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm [13, 14]. It has become somewhat ubiquitous, as JavaScript can now run not only on the browser, but on embedded systems, arduino chipsets, smart watches, etc. JavaScript was chosen since our system is intended to run on the web. It helped establish a bi-directional communication with the server.

The application is broken into two parts –front-end and back-end. The front-end was developed with HTML, CSS and JavaScript. The React library was used to handle the DOM (Document Object Model) and AJAX requests. The backend was developed using Node.js, which is a server-side framework that allows JavaScript to be written for server related matters [15]. Node runs on Google’s V8 engine. A lot of JavaScript modules were used in the project. Among them include: *Express* (This Node framework was used to handle the routing and parsing of requests using middleware more efficiently), *CSV-parser* (This was used to parse the csv dataset into a JSON for extraction and cleaning), *Jsonwebtoken* (This was used to generate the authentication). The *package.json* file contains a list of modules that were used.

System requirements

Details on system requirements are outlined in Table 4.

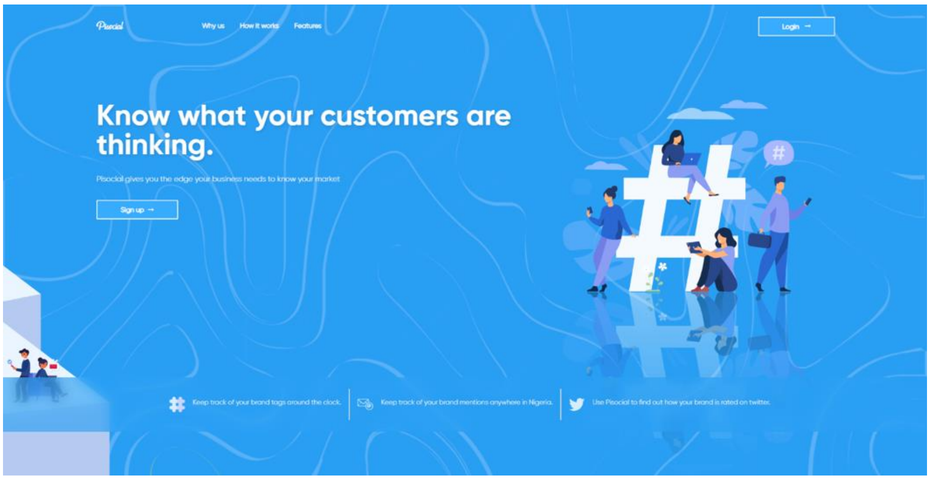
Table 4: System requirements

Requirements	**Details**
User Interface	Operating system: MacOS, Firefox OS, Linux, Windows XP, Windows Vista, Windows 7, Windows 8. HTML5 compatible for browser
Hardware	Web-based modules: • Processor: Pentium II or higher • Processor speed: 512MHz or higher • Hard disk: 20GB (minimum). • RAM: 512MB. • Monitor, mouse and keyboard, network interface card • Uninterrupted power supply (UPS) to avoid power failure (or a standby system).

Software	Operating system: MacOS, Firefox OS, Linux, Windows XP and later versions. Frontend: HTML5 compatible browser, JavaScript enabled browser. Backend: NodeJS and MongoDB

System development

This involves all processes that have been put in place to ensure success of the project. It is associated with all modules directly from the naming standards to the system engineering and afterward the advancement stage to the fruition of the venture. The system consists of modules or components working together. In this system, the discrete components are designed such that each component supports a well-defined abstraction. Figure 5a shows the first page on the system when visited, Figure 5b shows how Pisocial works, and Figure 5c shows the features available in Pisocial.



Know what your customers are thinking.

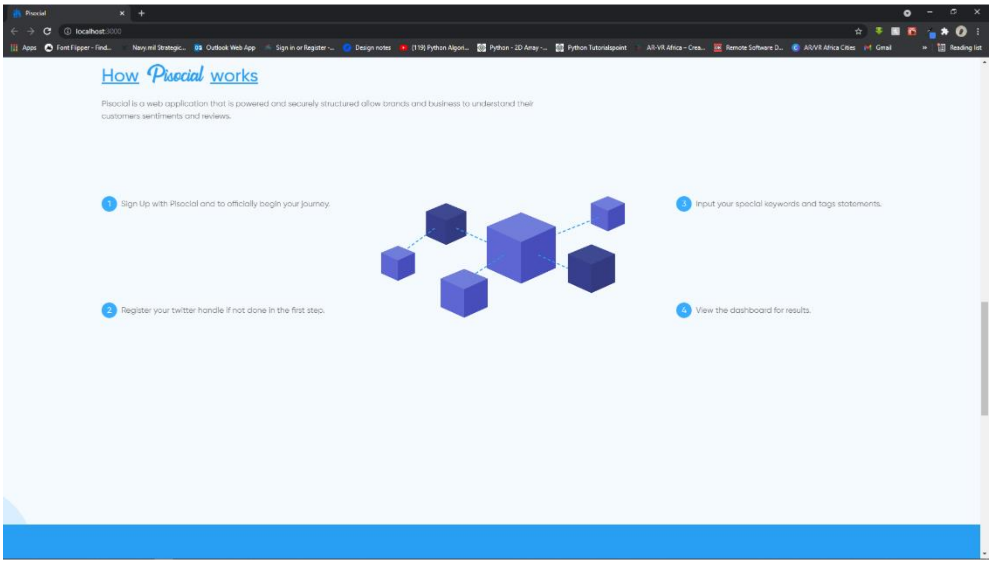
Pisocial gives you the edge your business needs to know your market

Keep track of your brand tags around the clock.

Keep track of your brand mentions anywhere in Nigeria.

Use Pisocial to find out how your brand is rated on twitter.

Figure 5a: Landing Page



How Pisocial works

Pisocial is a web application that is powered and securely structured allow brands and business to understand their customers sentiments and reviews.

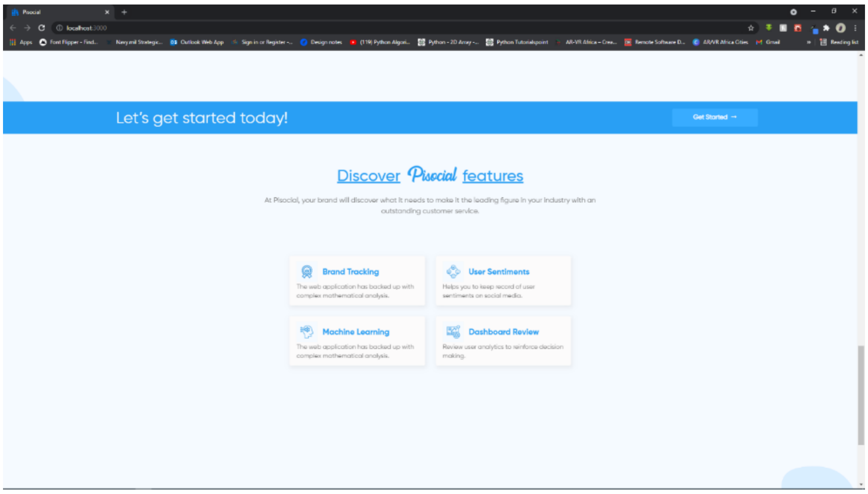
1 Sign Up with Pisocial and to officially begin your journey.

2 Register your twitter handle if not done in the first step.

3 Input your special keywords and tags statements.

4 View the dashboard for results.

Figure 5b: How it works



At Pisocial, your brand will discover what it needs to make it the leading figure in your industry with an outstanding customer service.

Brand Tracking The web application has backed up with complex mathematical analysis.

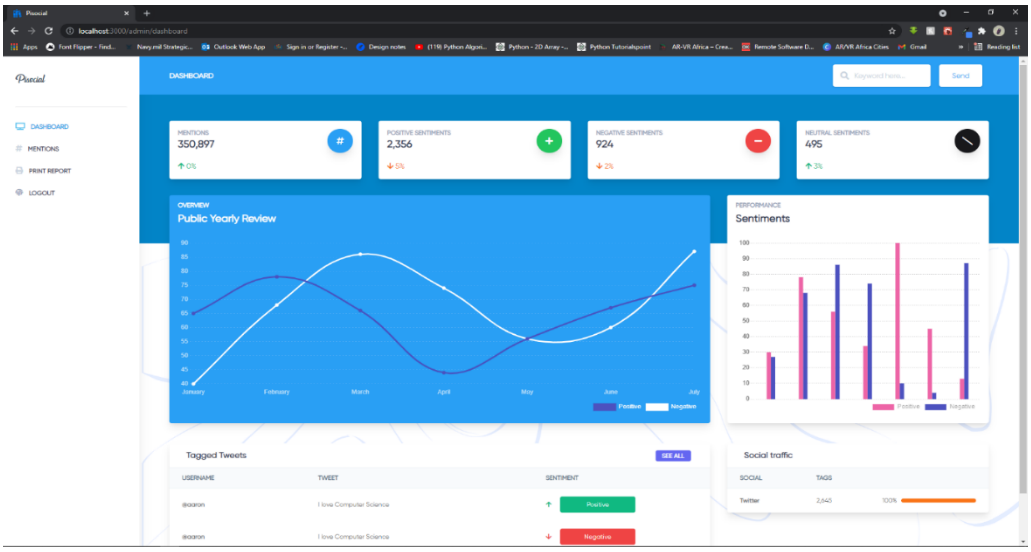
User Sentiments Helps you to keep record of user sentiments on social media.

Machine Learning The web application has backed up with complex mathematical analysis.

Dashboard Review Review user analytics to reinforce decision making.

Figure 5c: Features

The authentication and authorization modules used were necessary to protect and store user specific data. Without this module, illegal breaches into various companies' analysis and data may be inevitable. A structure was therefore put in place to make sure there are no breaches, and the system is secure. The admin module provides the functional components needed for the admin to operate in the system. This module contains the overview dashboard for the user to make informed decisions and to analyse client's sentiment towards the businesses. An option to generate a report is also created. This is facilitated using a sentiment analyser. Figures 6a and 6b show the dashboard page and the mentions page of the web application, respectively.



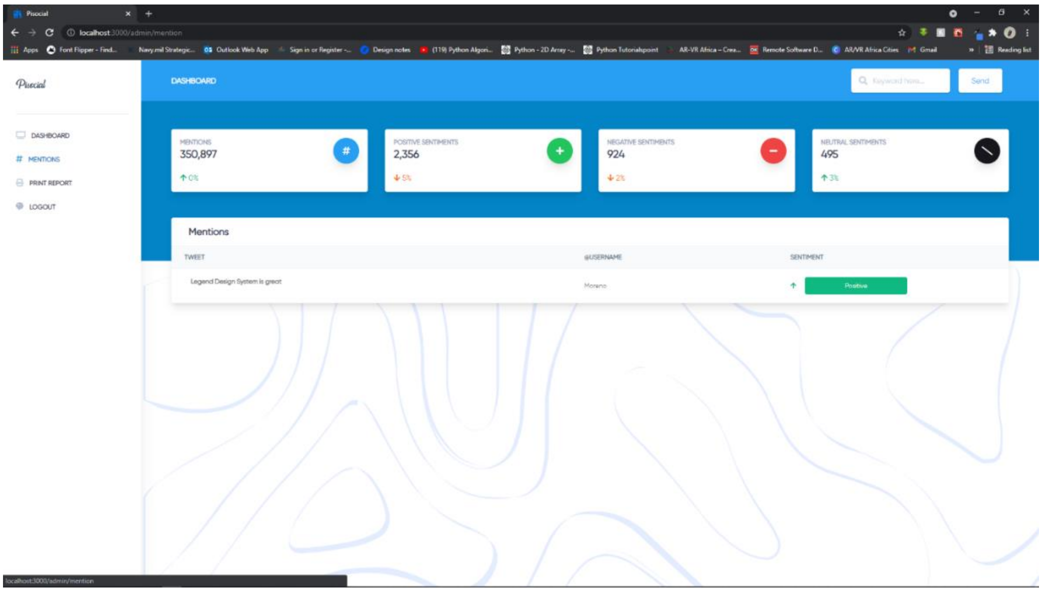
Tagged Tweets

USERNAME	TWEET	SENTIMENT
@aaron	I love Computer Science	↑ Positive
@aaron	I love Computer Science	↓ Negative

Social traffic

SOCIAL	TAGS	
Twitter	2,645	100%

Figure 6a: Dashboard



TWEET	@USERNAME	SENTIMENT
Legend Design System is great	Moreno	↑ Positive

Figure 6b: Mentions page

The sentiment analyser module helps to get the polarity of the twitter text that the 2016 US Presidential Election dataset offers. We make use of a JavaScript library called *Sentiment*. It makes use of a hybrid approach in solving natural processing language problems. It makes use of the AFFIN-165 dataset which classifies words based on their different sentiment. For example: cruel -3, cruelty -3, dirty -2, effective +2, peaceful +2, penalty -2, perfect +3. These are examples of how the AFFIN-165 dataset classifies its data. Sentiment uses a support vector machine (SVM) to classify the data in various points in space and separates them using a hyper-plane.

H: wT x + b = 0

This algorithm helps to further distinguish the polarity in the dataset. The accuracy for this library is suggested by validating its performance with datasets from Amazon and IMDb. Table 5 shows the accuracy of "Sentiment".

Table 5: Accuracy of *Sentiment*

	Amazon	IMDb
Accuracy	0.726	0.765

The accuracy of the system is calculated using the F-1 measure formula:

*F1-measure= (2 * (precision * recall) / (precision + recall))*

where precision = true positive / (true positive + false positive); recall = true positive / (true positive + false negative).

System testing

System testing of software or hardware is analysis conducted on a comprehensive, combined system to calculate the system's agreement with its stated requirements. (Rex, 2002). Testing encompasses finding problems in codes; debugging contains detecting and fixing problems. As a rule, system testing takes, as its input, all of the "included" software components that have effectively passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any irregularities between the software units that are integrated together and the system as a whole. Table 6 shows the types of tests used.

Table 6: System testing

Test	Description
Unit test	It is a strategy by which individual units of source code, sets of at least one PC program module along with related control information, utilization systems, and working strategies, are tried to choose if they are good for use. All the codes written down for the implementation of this system were properly tested, also all the modules were tested to ensure that the system meets its needs.
Integration test	It is a phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing [16]. Integration testing takes as its input modules that have been unit tested, groups them in larger collections, applies tests defined in an integration test plan to those collections, and delivers as its output the integrated system ready for system testing. All the modules of the proposed system were combined together and tested in order to produce the preferred output.

DISCUSSION

Pisocial, whose detailed documentation has been expounded, will perform for a lifetime with regular moderations. It makes use of a highly scalable NoSQL database called MongoDB. It relies on a fully extensive JavaScript backend built using the Node.js. framework. It is without doubt that the system has and will continue to effectively solve all the mentioned problems facing the economy whilst optimizing cost and preventing business losses. We believe this system will aid the populace in developing countries in solving all the problems stated in the *Introduction* section of this paper. This would ensure that businesses have within their reach all that are needed to have a better understanding of their clients, take better business decisions, know what niche to focus the most on, provide excellent services, take notes on how to create brand evangelists, and contribute significantly to the growth of their society. The capabilities of this system can be scaled far beyond the possibilities achieved in this paper. It can be further designed to handle the processes it tackles now even more seamlessly and efficiently.

CONCLUSION

Though this is a new system and might be prone to programming, logical or human errors, we recommend that it should be tested with a large populace in-order to get better sentiment data analysis. Doing so would provide the space, time and opportunity needed to contain all subsequent anomalies that might arise. Based on initial successful hosting and steady usage, this platform should be integrated for most businesses in developing countries.

This system can be improved upon in various ways. We intend to integrate it with social network giants, such as Twitter, although due to existing government policies, such integrations have been largely restricted. Successful integration will help fetch real time data from customers and their sentiments about the business. The system will be integrated with an effective inner system such that it can be called as a microservice in an external system. It will help businesses customize the features of the web app to be in line with what the business wants to achieve. We will also give an authorization hierarchy so that the app can be accessible to all business levels, but prevents modification depending on the user. Achieving this would allow for uniform information transfer in all levels, thereby leading to better organized data and analysis. There are plans for a mobile app for this platform for better access and convenience.

REFERENCES

1. CURRAN, K., O'HARA, K. & O'BRIEN, S. (2011). The role of twitter in the world of business. *International Journal of Business Data Communications and Networking*, **7**(3): 1-15.
2. ALI TAHA, V., PENCARELLI, T., ŠKERHÁKOVÁ, V., FEDORKO, R. & KOŠÍKOVÁ, M. (2021). The Use of Social Media and Its Impact on Shopping Behavior of Slovak and Italian Consumers during COVID-19 Pandemic. *Sustainability*, **13**: 1710. https://doi.org/10.3390/su13041710
3. GRUEN, T.W., OSMONBEKOV, T.D. & CZAPLEWSKI, A.J. (2006). eWOM: The impact of customer-to-customer online know-how exchange on customer value and loyalty. *Journal of Business Research*, **59**(4): 449-456
4. WILSON, A. (2015). *Why 50% of online shoppers will check your social media before buying*. information on https://www.linkedin.com/pulse/why-50-online-shoppers-check-your-social-media-before-arthur-wilson
5. MACMILLAN, A. (2020). How empathy, ahead of tech, will be the key driver of business recovery post-coronavirus, *Digital marketing magazine*, information on https://digitalmarketingmagazine.co.uk/
6. TARTAGLIONE, A.M., CAVACECE, Y., GIUSEPPE, R. & GRANATA, G. (2019). A systematic mapping study on customer loyalty and brand management. *Administrative Sciences*, **9**(1): 1-21.
7. HUDSON, S., ROTH, M.S., MADDEN, T.J. & HUDSON, R. (2015). The effect of social media on emotion, BRQ and WOM: An empirical study of music festival attendees. *Tourism Management*, **47**: 68-76. doi:10.1016/j.tourman. 2014.09.001
8. SINGLA, Z., RANDHAWA, S. & JAIN, S. (2017). Sentiment analysis of customer product reviews using machine learning. In Proceedings of the International Conference on Intelligent Computing and Control (I2C2), Coimbatore, India, 23–24 June 2017, pp. 1–5.
9. APPEL, G., GREWAL, L., HADI, R. & STEPHEN, A.T. (2020). The future of social media in marketing. *Journal of the Academy of Marketing Science*, **48**: 79-95. https://doi.org/10.1007/s11747-019-00695-1.
10. MVC information on https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller

11. HERNANDEZ, R.D. (2021), The Model View Controller Pattern – MVC Architecture and Frameworks Explained, information on https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained/
12. TUTORIALSPOINT, MVC *Framework – Introduction*, information on https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm
13. RAUSCHMAYER, A. (2019). *JavaScript For Impatient Programmers* (ES2022 edition), 526p, Independently published ISBN 1091210098, information on https://exploringjs.com/impatient-js/downloads/impatient-js-preview-book.pdf
14. TUTORIALSPOINT, JAVAScript, information on https://www.tutorialspoint.com/javascript/javascript_tutorial.pdf
15. SHAH, H. (2017). Node. js challenges in implementation. *Global Journal of Computer Science and Technology*, 17.
16. WEISS, D. & ZDUNIAK, M. (2007). Automated Integration Tests for Mobile Applications in Java 2 Micro Edition. In: Abramowicz, W. (eds) *Business Information Systems*. BIS 2007. Lecture Notes in Computer Science, vol 4439. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-72035-5_37.