

PREDICTION OF EDUCATIONAL DATA USING DEEP CONVOLUTIONAL NEURAL NETWORK

Ms.K. VIJAYALAKSHMI, II ME CSE - Hindusthan College of Engineering and Technology
Ms. M. RAMYA DEVI M.E., (Ph.D.), Assistant Professor, CSE - Hindusthan College of Engineering and Technology
Dr.S.Uma, Research coordinator, , CSE - Hindusthan College of Engineering and Technology

Abstract

Sentiment analysis is one of the utmost active research areas in natural language processing, web mining and text mining. In education, big data is a significant research part used to progress the value of education by observing students' performance and understanding their learning patterns. Real-time student feedback would empower faculty and students to comprehend the teaching and learning problems in the most user-friendly way for the students. The proposed Deep Convolutional Neural Network (DCNN) uses a Lexicon based sentimental analysis technique to analyze students' feelings and emotions through their feedback by correlating learning analytics to grounded theory. The sentiment analysis technique is a computational process to identify and classify subjective information such as positive, negative, and neutral from the source material. The proposed DCNN to improve the quality of teaching by giving positive or negative sentiments. This proposes a model based on Deep Learning approach to perform sentiment analysis

INTRODUCTION

Live chat sentiment analysis is a feature that analyses every chat session for positive and negative mood indicators. Contact Lens lets Amazon Connect users glean insights from contact center support calls and chats by using machine learning and AI to analyze those conversations on the spot. It eats into the time agents could spend on other calls, reducing efficiency and call volume. Advancements in technology continue to provide organizations with opportunities to enhance their online service delivery. Many organizations are now offering customer service and online support through instant messaging platforms, known as 'live chat analysis' systems. These services allow customers to seek service related information from an organization via online-based synchronous media and a human service representative who provides answers through such media. Despite the environment, whether it be online or offline, organizations realize the importance of high quality customer service. Therefore, in an attempt to provide effective online customer service, online helpdesks and live chat

functions are being adopted as customer service platforms.

Sentiment analysis on the live chat video comments is a process of understanding, extracting, and processing textual data automatically to obtain sentiment information contained in one sentence of live chat video comment. Text mining approach becomes the best alternative to interpret the meaning of each comment. The classification of positive and negative content becomes very important for the live chat user to assess how meaningful the content that has been published is based on user opinion is extensively used as a basic line in tasks related to texts but the performance varies significantly in all variants, features, and numbers of data

collection.

While live chat services have numerous potential advantages, the success of such facilities depend on the experience encountered during use. Typically, customers have various options when it comes to seeking customer service assistance, including face-to-face, telephone, social media and email. In spite of this, many customers now prefer to use online live-chat facilities for service related questions such as inquiries about products, orders, shipping options and access to information. Despite the usefulness of online live chat systems in enhancing service experience, research to further our understanding of the dynamics and influencing factors of this concept is scarce in the extant literature. Thus, understanding the variables capable of influencing a customer's service encounter with a live chat operator becomes exceptionally important.

Information System Success model, this research explores the variables capable of influencing a customer's satisfaction with their experience during an online live chat service encounter with a human service representative. To our knowledge, there are no empirical studies that examine the variables influencing satisfaction with the live chat experience, despite the growing number of organisations and customers adopting live chat as a customer support function. As a result, this study aims to fill such a gap in knowledge by providing an empirical perspective to further our understanding of online customer support with regard to online live chat systems. We therefore, incorporate key facets from both Service Marketing literature and Information Science

literature and draw upon Information Systems Success model to help gain an understanding of customer perceptions of web-based support. Thus, being the first to explore service, information and technological variables influencing a live chat service encounter in the varying purposes of use, along with specific live chat features, this research makes a significant contribution to theory by providing essential insights that extend our understanding on online

customer support services in the form of live chat facilities. The following section will discuss the conceptual background.

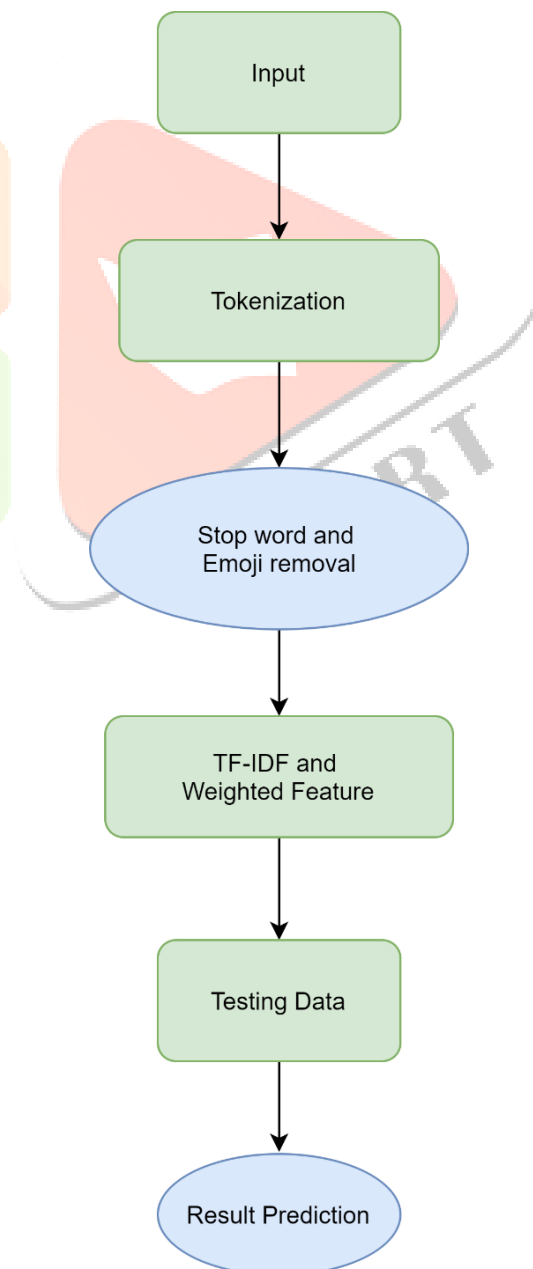
Service quality and its measures have received much attention over the past number of years outline service quality dimensions as proposed by as important indicators in assessing the service quality of an information system, namely, reliability, assurance, empathy and responsiveness. Reliability refers to the consistency of performance and dependability of the service provider. Assurance involves competence, courtesy, credibility and security. Responsiveness refers to the willingness and readiness of service staff to provide the service in a timely manner. Lastly, Empathy refers to the level of customer understanding and communication provided by service.

The service representative used some automated responses for frequently asked questions and greetings with the live chat system. Due to the range of live chat systems that were used in their natural setting for true customer support enquiries we are able to produce generalizable results. In responses were collected with usable questionnaires, which is an adequate sample size for structural equation modelling with analysis of moment structures. The benefit of structural equation modelling is that the hypothesized model can be tested simultaneously in an analysis of the model of variables.

An online questionnaire was used in order to capture the data required to test the hypothesized relationships. Data was collected from individuals in the used a mobile phone network provider's web-based live chat facility within 5 days of taking the survey. Mobile phone networks have adopted web based live chat facilities as one of the main methods of seeking customer support, therefore they offered an interesting context to study. Each mobile network's website used by the respondent was captured. Respondents were purposively selected to participate in the study. As a non-probability sampling technique, this procedure is criticized for

its subjectivity. However assert that a purposive sample apart from being representative is argued to be convenient, requires fewer resources (cost and time), and is as good as probability sampling. A pilot study was conducted with a sample of respondents prior to collecting the data to assess the logic and design of the questionnaire. The majority of questionnaire scales were adapted from established scales within the literature to measure Reliability, Assurance, Empathy, Responsiveness, Perceived Wait Time, Perceived Information Quality, Perceived Usefulness, Perceived Ease of Use, and Satisfaction with the Experience. Four new scales were introduced to measure, attitudes towards the service rep's picture, system response time estimations, use of emoticons and automated 'canned' responses.

System Diagram



SYSTEM STUDY

Existing System

Dataset is a simple text file that has been extracted from any of the live chat groups or one to one individual chat. More the number of text messages, the more the accuracy will be in identifying the sentiment. The live chat can be extracted using a feature called export chat and this will mail the compressed has a text file of the chat from the beginning and all the undeleted chat will be included in this text file. A lot of pre-processing needs to be done used a method named Natural Language Process (NLP).

Disadvantages

It's harder to keep a conversation flowing live chat than it is over the phone support, or even email, which may make the interactions less efficient.

- Lack of job security.

MODULES DESCRIPTION

Data Preparation

Data preparation is the process of cleaning and transforming raw data prior to processing and analysis. It is an important step prior to processing and often involves reformatting data, making corrections to data and the combining of data sets to enrich data.

Data set load

The Load method provides a technique for filling a single Data Table with data, retrieved from a Data Reader instance. This method provides the same functionality but allows you to load multiple result sets from an IDataReader into multiple tables within a Dataset.

Data Pre-processing

Data preprocessing is an important step in the data mining process. The phrase "garbage in, garbage out" is particularly applicable to data mining and machine learning projects. Data-gathering methods are often loosely controlled, resulting in out-of-range values, impossible data combinations, and missing values.

- Privacy problems have emerged

Proposed system

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format. Steps Involved in Data Preprocessing to handle this part, data cleaning is done. It involves handling of missing data, noisy data. Data virtualization is an approach to data management that allows an application to retrieve and manipulate data without requiring technical details about the data, such as how it is formatted at source. A Deep Convolutional Neural Network is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence.

Advantages

- Training chatbots
- Identifying key emotional triggers
- Live insights

Data virtualization and natural language process (NLP) sentimental analysis

Natural language processing (NLP) is the relationship between computers and human language. Natural language refers to speech analysis in both audible speech, as well as text of a language. NLP systems capture meaning from an input of words.

Deep convolutional neural network using classification

A Deep Convolutional Neural Network is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. Deep convolutional neural networks (CNN or DCNN) are most used to identify patterns in images and video. Deep convolutional neural networks are mainly focused on applications like object detection, image classification, recommendation systems. A deep CNN can learn basic filters automatically and combine them hierarchically to enable the description of latent concepts for pattern recognition

SYSTEM DESIGN

Introduction

The System Design Document describes the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces.

Executive summary of the project

This section provides a description of the project from a management perspective and an overview of the framework within which the conceptual system design was prepared. If appropriate, include the information discussed in the subsequent sections in the summary.

Process of System overview

This section describes the system in narrative form using non-technical terms. It should provide a high-level system architecture diagram showing a subsystem breakout of the system, if applicable. The high-level system architecture or subsystem diagrams should, if applicable, show interfaces to external systems. Supply a high-level context diagram for the system and subsystems, if applicable. Refer to the requirements traceability matrix (RTM) in the Functional Requirements Document (FRD), to identify the allocation of the functional requirements into this design document.

Constraints on the design

This section describes any constraints in the system design (reference any trade-off analyses conducted such, as resource use versus productivity, or conflicts with other systems) and includes any assumptions made by the project team in developing the system design.

Planning for software design

A software module is the lowest level of design granularity in the system. Depending on the software development approach, there may be one or more modules per system. This section should provide enough detailed information about logic and data necessary to completely write source code for all modules in the system (and/or integrate COTS software programs).

If there are many modules or if the documentation is extensive, place it in an appendix or reference a separate document. Add additional diagrams and information, if necessary, to describe each module, its functionality, and its hierarchy. Industry-standard module specification practices

should be followed. Include the following information in the detailed module designs:

A narrative description of each module, its function(s), the conditions under which it is used (called or scheduled for execution), its overall processing, logic, interfaces to other modules, interfaces to external systems, security requirements, etc.; explain any algorithms used by the module in detail

For COTS packages, specify any call routines or bridging programs to integrate the package with the system and/or other COTS packages (for example, Dynamic Link Libraries) Data elements, record structures, and file structures associated with module input and output

Graphical representation of the module processing, logic, flow of control, and algorithms, using an accepted diagramming approach (for example, structure charts, action diagrams, flowcharts, etc.)

Data entry and data output graphics; define or reference associated data elements; if the project is large and complex or if the detailed module designs will be incorporated into a separate document, then it may be appropriate to repeat the screen information in this section.

Process of Input design

Input Screen must be design in such a way to give an easy navigation throughout the screen without the violation of the input validation. Input design is the process of converting the user-originated data into a computer-based format. Inaccurate input data are the most common cause of error in data processing. The goal of an input data are collected and organized into a group and error free. Input data are collected and organized into a group of similar data. Once identified, appropriated input media are selected for processing. The design was done with six major objectives in mind

- Effectiveness
- Accuracy
- Ease of Use
- Consistency
- Simplicity
- Attractiveness

The main objective of designing input focuses on: Controlling the amount of input required

- Avoiding delayed response
- Controlling errors
- Keeping process simple
- Avoiding errors
- Producing cost effective method of input.
- Achieving highest possible level of accuracy.

Ensure that the input is acceptable to and understood by the staff.

The goal of designing input data is to make entry easy, logical and free from errors as possible. The entering data entry operators need to know the allocated space for each field, field sequence and which must match with that in the source document. The processor analyzes the input required. It is then accepted or rejected.

The fundamental goal of planning input centers on:

- Controlling the amount of input required
- Avoiding delayed response
- Controlling errors
- Keeping process simple
- Avoiding errors
- Delivering savvy technique for input.
- Accomplishing the most elevated conceivable degree of exactness.
- Guarantee that the information is satisfactory to and perceived by the staff.

Enter the plan objective is to make the information section simple and legitimate as conceivable from errors and opportunity. In entering the information passage, the administrator has to realize each field's space, the field dispersion of the request, and source documents should coordinate. The processor breaks down the information required at that point, and it is acknowledged or dismissed.

The main objective of designing input focuses on

- Controlling the amount of input required
- Avoiding delayed response
- Controlling errors
- Keeping process simple
- Avoiding errors
- Producing cost effective method of input.
- Achieving highest possible level of accuracy.

Ensure that the input is acceptable to and understood by the staff.

The goal of designing input data is to make entry easy, logical and free from errors as possible. The entering data entry operators need to know the allocated space for each field, field sequence and which must match with that in the source document. The processor analyzes the input required. It is then accepted or rejected.

Output Design

The normal procedure in developing a system

is to design the output in detail first and then move back to the input. The output will be in the form of views and reports. The output from the system is

required to communicate the result of processing to the users. They are also used as the permanent copy for later verifications.

Output Design consideration

The purpose of outputs has been understood and the efficiency of information contained should be analyzed and confirmed. Then the output have been defined in terms of

- Name of the Output
- Content
- Format
- Frequency

Outputs

This section describes of the system output design relative to the user/operator; show a mapping to the high-level data flows described in Section. System outputs include reports, data display screens and GUIs, query results, etc. The output files are described in Section 3 and may be referenced in this section. The following should be provided, if appropriate:

- Identification of codes and names for reports and data display screens
- Description of report and screen contents (provide a graphic representation of each layout and define all data elements associated with the layout or reference the data dictionary)
- Description of the purpose of the output, including identification of the primary users
- Report distribution requirements, if any (include frequency for periodic reports)
- Description of any access restrictions or security considerations

Code Design

A design pattern is a

standardized solution to a software design issue or problem which is encountered daily in real-world application development. A pattern focuses on class design and object interaction. Knowledge of design patterns not only prevents having to re-invent the wheel, it allows developers to discuss their work at a higher level of abstraction.

Design patterns have been the bane of my programming existence. I have trouble learning and remembering them. On the one hand, I feel like I have always been following such patterns throughout my career—even before object-oriented languages. On the other hand, I haven't been able to get a good enough handle on patterns and the terminology to be able chat freely about them with my colleagues

where a longer and more targeted sales pitch may lose the offender's interest.

Database design

The database design involves creation of tables that are represented in physical database as stored medical information. They have their own existence. Each table constitute of rows and columns where each row can be viewed as record that consist of related information and column can beviewed as field of data of same type.

Framework for system Design

The design effort transforms the detailed, defined requirements into complete, detailed specifications that direct development and testing. Design decisions detail how the system will meet the defined functional, physical, interface, security, and data requirements. At the end of the design process the design is base lined.

The general system characteristics are defined during design. The operating system is established and the automated system packaged into major design subsystems. Inputs and outputs of each subsystem are defined, interfaces to external systems are designed, and administrative activities are established. Security and auditing needs are also addressed.

A more detailed structure of the system is then created based on the subsystems identified by the general characteristics. Each subsystem is partitioned into one or more design units, or modules. The process is described in a structure chart, flowchart, action diagram, pseudo code, or other acceptable format for each design unit, or module. Detailed logic specifications are written for each module described and data usage is physically defined to the elemental level.

Functions requiring user input and approval are completed in this activity.

Throughout the design phase there are a series of check point and review processes. The design is reviewed to verify that it has the following characteristics:

- Is directly traceable to the requirements.
- Describes how the capabilities defined by the requirements will be implemented.
- The SDD includes
 - User, human/computer interface design
 - System architecture
 - Detailed system design
 - Data base design including a physical data model and data dictionary.

SYSTEM TESTING

Before applying method to design effective test cases, a software engineer must understand the basic principles that guide software testing. Davis (DAV95) suggests a set of testing principles which have been adapted for use in this book.

- All tests should be traceable to customer requirements.
- Test should be planned long before testing begins.
- Tests pare to principle applets to software testing. Testing should begin “in the small” and progress towards testing “in the page”.
- Exhaustive testing is not possible.

TESTING STEPS

- Unit testing
- Integration Testing
- Whitebox Testing
- Acceptance testing
- Alpha Testing
- Beta Testing
- Blackbox Testing

UNIT TESTING

Unit testing focuses on verification errors on the smallest unit of software design- the module. Using the procedural design description as a guide, important control paths are tested to uncover errors within the boundary of the module.

The module interface is tested to ensure that the information properly flows into and out of the program unit under test. Boundary conditions are tested to ensure that the module operates properly at the boundaries established to limit of restrict processing.

Integration test

Integration testing is a systematic technique for constructing the program structure while conducting test to uncover errors associated with interfacing. The objective is to take unit tested modules and build a program structure that has been dictated by design.

Whitebox testing

White box testing is some time is called glass box testing, is a test case design that uses a control structure of the procedural design to drive the test cases. Using white-box testing methods, the software engineer can drive test cases that

- Guarantee that logical decisions are on the true and false sides
- Exercise all logical decisions are on the true and false sides
- Execute all loops at their boundaries and within their operational bounds
- Exercise internal data structure to assure the validity

Acceptance testing

Finally, when the software is completely built, a series of acceptance tests are conducted to enable the client to validate all requirements. The user conducts these tests rather than the system developer, which can range from informal test drive to a planned and systematic executed series of tests. These acceptance tests are conducted over a period of weeks or months, there by uncovering cumulative

errors that might degrade the system order time. In this process alpha testing and beta testing are used to uncover the errors that only the end user seems able to find.

Alpha testing

The customer conducts the alpha test at the developer’s site. The client notes the errors and usability problems and gives report to the developer. Alpha tests are conducted in a control environment.

Black box testing is not an alternative for white box testing techniques. Rather, it is a complementary approach that is likely to uncover different class of errors. Black box testing attempts to find errors in the following categories:

- Interface errors.
- Performances in data structures or external database access.
- Performance errors.
- Initialization and termination errors.
- Incorrect or missing functions.

APPENDIX 2 Screen shots

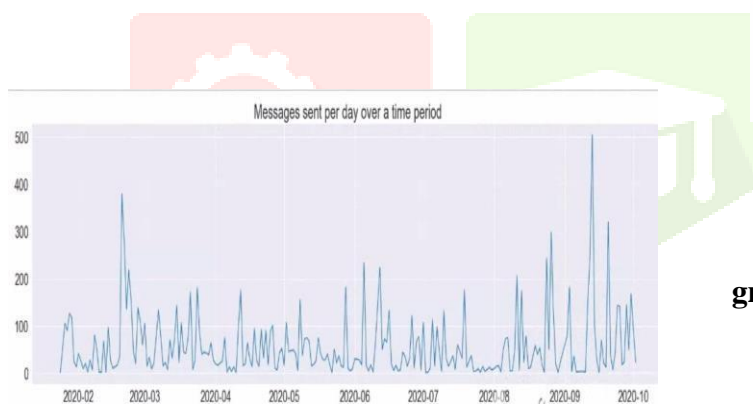


Figure A.1.1 Message sent per day over a time period

Black Box Testing

Black box testing focuses on the functional requirements of the software. That is black box testing enables the software engineer to drive a set of input conditions that will fully exercise the requirements for a program.

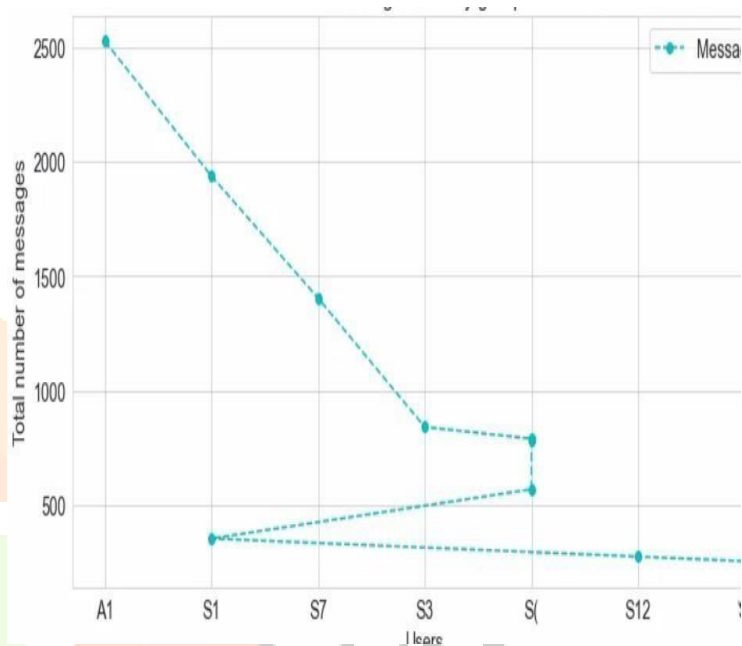


Figure A.1.2 Number of Message sent by group

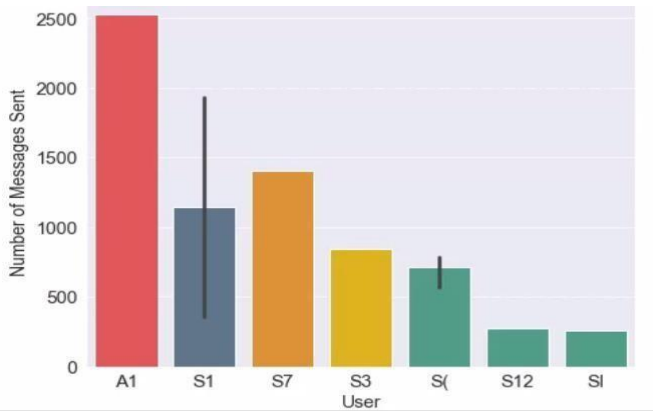


Figure A.1.3. Total Message sent

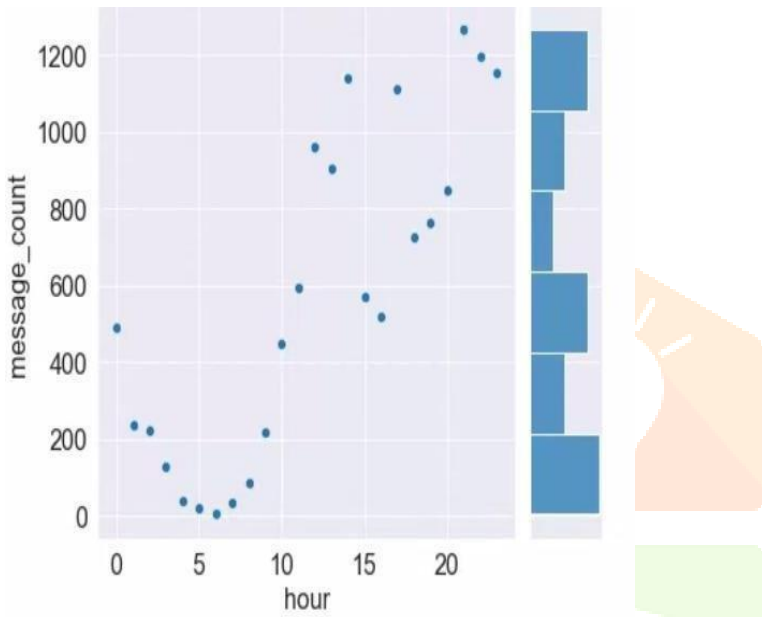


Figure A1.4. Most active user

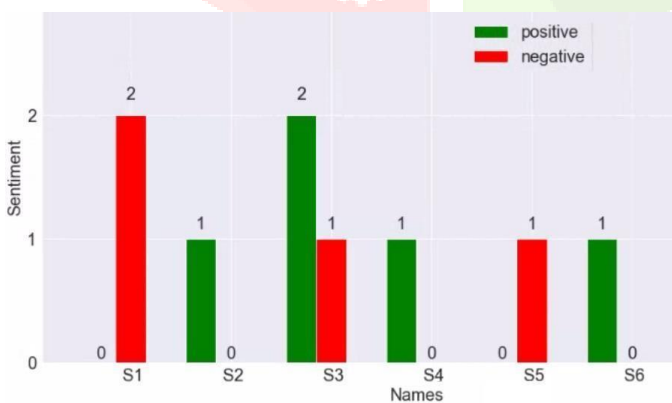


Figure A.1.5 Sentiment analysis

Conclusion

Live chat will be able to chat classification using Deep Convolutional Neural Network using the performance. Live chat is ideal for phone calls and e-mail. That visitors need to do is to send enter your e-mail address in the chat box. If you are using, interview web-based survey, is the actual, including increasing the experience of real-time chat service, and the presence of the network provider of mobile phone to adopt the whole of the structural equation model between the Natural Language Processing (NLP). The proposed algorithm using DCNN for the classification of online live chat service. As a result, preprocessing data and data analysis for good sentimental analysis performance for Deep Convolutional Neural Network performance.

Reference

- [1] H. Aizawa, S. Iwasaki, R. Gomi, E. Sato-Shimokawara and T. Yamaguchi, "Heart rate analysis in a conversation on video chat for development of a chat robot supporting to build a relationship," 2017 IEEE/SICE International Symposium on System Integration (SII), Taipei, Taiwan, 2017, pp. 740-745, doi:10.1109/SII.2017.8279310.
- [2] H. Minami, H. Kawanami, M. Kanbara and N. Hagita, "Chat robot coupling machine responses and social media comments for continuous conversation," 2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Seattle, WA, USA, 2016, pp. 1-6, doi: 10.1109/ICMEW.2016.7574758.
- [8] E. Sato-Shimokawara, Y. Shinoda, T. Takatani, H. Lee, K. Wada and T. Yamaguchi, "Analysis of category estimation for cloud based chat robot," 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), New York, NY, USA, 2016, pp. 308-311, doi: 10.1109/ROMAN.2016.7745147.
- [9] C. -M. Liaw and B. -R. Dai, "Live Stream Highlight Detection Using Chat Messages," 2020 21st IEEE International Conference on Mobile Data Management (MDM), Versailles, France, 2020, pp. 328-332, doi: 10.1109/MDM48529.2020.00072.
- [10] A. Elmorshidy, "Benefits Analysis of Live Customer Support Chat in E-commerce Websites: Dimensions of a New Success Model for Live Customer Support Chat," 2011 10th International Conference on Machine Learning and Applications and Workshops, Honolulu, HI, USA, 2011, pp. 325-329, doi: 10.1109/ICMLA.2011.167.
- [3] M. Skjuve and P. B. Brandtzaeg, "Facebook Live: A Mixed-Methods Approach to Explore Individual Live Streaming Practices and Motivations on Facebook," in *Interacting with Computers*, vol. 31, no. 3, pp. 589-602, May 2019, doi: 10.1093/iwc/iwz038.
- [4] Y. Zhao, A. Zhou and X. Chen, "Reducing Latency in Interactive Live Video Chat Using Dynamic Reduction Factor," 2020 IEEE Wireless Communications and Networking Conference (WCNC), Seoul, Korea (South), 2020, pp. 1-6, doi: 10.1109/WCNC45663.2020.9120837.
- [5] A. A. N. Buker, G. Roffo and A. Vinciarelli, "Type Like a Man! Inferring Gender from Keystroke Dynamics in Live-Chats," in *IEEE Intelligent Systems*, vol. 34, no. 6, pp. 53-59, 1 Nov.-Dec. 2019, doi: 10.1109/MIS.2019.2948514.
- [6] K. Hirata, E. Shimokawara, T. Takatani and T. Yamaguchi, "Filtering method for chat logs toward construction of chat robot," 2017 IEEE/SICE International Symposium on System Integration (SII), Taipei, 2017, pp. 974-979, doi: 10.1109/SII.2017.8279349.
- [7] N. C. Nguyen, Z. Wei, P. Paliyawan, H. V. Pham, R. Thawonmas and T. Harada, "Using GWAP to Generate Informative Descriptions for Artwork Images on a Live Streaming Platform," 2019 IEEE International Conference on Consumer Electronics - Asia (ICCE-Asia), Bangkok, Thailand, 2019, pp. 43-44, doi: 10.1109/ICCE-Asia46551.2019.8941600.