Customer Critique Analysis System for PT. KCI’s Twitter

Ahmad Husen, Sari Widya Sihwi, Esti Suryani
Informatics Department, Fakultas MIPA
Universitas Sebelas Maret
Jl Ir. Sutami 36A Surakarta
sariwidya@staff.uns.ac.id

Abstract. Commuter Line is one of the main choices for users of public transport in Jakarta. Commuter Line in Indonesia is managed by PT. Kereta Commuter Indonesia (PT. KCI). PT. KCI has an official Twitter account @CommuterLine. The purpose of this research was built Customer Critique Analysis System for Twitter PT. Kereta Commuter Indonesia using Support Vector Machine (SVM). The software engineering method used in this research is Component-Based Software Engineering (CBSE). Applications built in this research uses some components/libraries, which are Tweetsharp, LibSVM, Metronic, Kendo UI and JQCloud. The results of this research is a useful analytics application to perform grabbing and tweet data analysis from Twitter account @CommuterLine, and a dashboard to display the results of the analysis in graphical form. The test results show that the system meets the functional and non-functional requirements from the user. Testing is done by installing the application on the server PT. KCI, then it was tested by user. Non-functional testing was done by measuring application performance and usability. Performance measured is classification accuracy and running time of the program. Accuracy performance testing resulted in average sentence classification accuracy of 80.4%, and the accuracy of complaint classification type of 82.3%. The average running time is 3 minutes 25 seconds.

1. Introduction
Electric Railway or Commuter Line (KRL) is one of the main public transportation in Jakarta. This is because Commuter Line is a transportation that is free from congestion and has a very affordable rates. Commuter Line is operated by PT. Kereta Commuter Indonesia (PT. KCI), a subsidiary of the Indonesian national railway company PT Kereta Api Indonesia (KAI). According to the data (until June 2018) from the official website of PT. KCI, the average number of KRL users per day reached 1,001,438 users on weekdays, with the highest number of users being served in one day was 1,154,080 [1].

As a main public transportation, PT. KCI continues to improve its services and facilities to satisfy passengers to remain being a top choice of transportation. As a result, PT. KCI collects all passengers’ complaints and feedbacks from its official Twitter account named @CommuterLine and evaluates all of them regularly. The passengers are very enthusiastic with this, as can be seen in January - July 2017 the average tweets that go into PT. KCI’s Twitter account is 543 Tweets / Day . In order to evaluate regularly the big number of data manually is a big deal for PT. KCI.
For this reason, a kind of combination of big data analytic (BDA), which is customer critique analysis system is needed to help PT. KCI. Customer critique analysis system is a type of BDA focusing in critiques or complaints. It is still an interesting topic to be issued since not many of railway-related enterprises have implemented BDA in one or more railway transportation system areas [2]. In the transportation domain, there are some researches that also use passengers’ tweets to improve their services, for examples research [3] and research [4]. Similar with other majority of the researches, they also stopped until analyzing the polarities of passengers’ comments and without building an interface that can be used continuously.

Therefore, this research aims to build a customer critiques analytics system that has the the ability to accomplish two tasks. First, it has to grab the tweets and filter them from any tweets that do not contain critiques or feedbacks. Second, this system has to process the tweets and display them in an effective way so PT. KCI can get enormous information to improve its services.

In this case, the customer critiques analytics system should have the ability to accomplish two tasks. First, it has to grab the tweets and filter them from any tweets that do not contain critiques or feedbacks. Second, this system has to process the tweets and display them in an effective way so PT. KCI can get enormous information to improve its services.

To filter all the tweets and gain the tweets containing critiques and feedbacks, sentiment analysis can be a solution. Sentiment analysis is a type of data mining that using natural language processing (NLP) techniques and computational intelligence models to analyze subjective information from sources such as social networks in order to determine the polarity of people’s opinions about someone or something [5]. There are three types of polarity, which are positive, negative and neutral [6]. In this case, the system has to identify all the tweets to all types of polarity and takes only the negative one to be analyzed, since the negative polarity always associate with critiques or feedbacks.

Algorithm that is used in this research is Support Vector Machines (SVM), which is one of powerful algorithms that can be used to do a sentiment analysis. SVM is a machine learning approach that find a maximum margin function that separate a large set of observations with known labels into two classes where each observation is a point in a multidimensional space of feature measurements. One of the advantages of SVM is its ability to handling sparsity which refers to the ability to express the classification function in terms of a relatively small portion of the training data [7]. SVM has been implemented in many researches that using Indonesian text as the data and the results are very satisfying. Few examples of them are researches [8],[9], [10] reached accuracy values as follows, which are 95.77%, 90.73, and 87%.

To display passengers’ critiques in the forms of something that easiest to analyze and to be used in decision making, a dashboard can be a solution. Dashboard is a tool developed in the business sector that firstly used to recapitulate and integrate key performance information across an organization into a visual display that can help to do an operational decision making [K.Pauwels, et al in [3]]. There are some researches that also built dashboard in many domains as their final goals. Some of them are [12] that used a dashboard as a policy strategy vehicle of i-cities in Stockholm, [13] that build a dashboard to be a complete decision support framework for sentiment analysis and [14] that build a dashboard that support the decision-making process of academic advisers through comparative and predictive analysis. In this research, the dashboard built are based on PT. KCI’s requirements and business’ needs.

2. Research Methodology
Research methods that will be used in this research is Reuse-Oriented or Component Based Software Engineering. The reuse-oriented approach relies on the base of reusable software components and these components will be reintegrated with the framework [15]. The steps that was done in this research can be seen in Figure 1.
A. Requirement Specification
   The Requirement Specification phase was done using interview method with the user that is PT. KCI.

B. Component Analysis
   The Component Analysis phase was done to find the components that match the needs derived from the requirement specification stage.

C. Requirement Modification
   In the Requirement Modification phase, all functions are determined to be used from the components obtained in the Component Analysis stage.

D. System Design with Reuse
   In the System Design with Reuse phase, the application framework to be built was designed using the function of the components that have been obtained.

E. Development and Integration
   In this phase, the design result was implemented using Visual Studio 2015 Community Edition and SQL Server 2014.

F. System Validation
   The application resulted in previous phase was tested by the user in this phase. A performing tests was done to check whether the functions meet the requirements or not. After that is done accuracy testing by using data that has been labeled manually, then compared with the results of the classification by the application. Prior to testing, it takes preparation of training data to be used by the application.

3. Result and Discussion

3.1. Requirement Specification
   The initial stage in this step is analyzing the needs of end user. Method used in this research is interview that was done by interviewing Supervisor of Customer Relationship Division from PT. KCI. All the results can be divided into two group, which are Functional Requirement and Non-Functional Requirement. The functional requirement required by the system can be seen in table 1. The non-functional requirements required by this system is related with Performance. Performance types that
need to be measured are the classification accuracy using SVM and the processing time of the system
starting from preprocessing until the completion of classification.

Table 1. The list of Functional Requirements

| Code | Description |
|------|-------------|
| FR1  | The system can retrieve data H-1 tweet from twitter account PT. KCI automatically every day. |
| FR2  | The system can perform sentiment analysis of the data obtained to separate positive, negative and neutral sentiment categories. |
| FR3  | The system can classify the types of complaints from tweets that are negative, the types of complaints are divided into six types of complaints, which are Queues and Delay, Capacity, Cleanliness, Facility, Services and Security |
| FR4  | The system can classify Commuter Line route from tweet data obtained based on the station's keyword, which are Central Line, Loop Line, Bekasi/Cikarang Line, Bogor/Depok Line, Tangerang Line, Serpong Line |
| FR5  | The system can display the percentage of complaint types in graphical form. |
| FR6  | The system can display Twitter users who post the most criticisms to the Commuter Line account in graphical form. |
| FR7  | The system can display 5 most popular criticism tweets based on the number of retweet in tabular form. |
| FR8  | The system can display the words that most often appear from the tweet data obtained and displayed in word cloud form. |
| FR9  | The system can display a graph of daily and monthly tweets based on the type of sentiment. |
| FR10 | The system displays all the graphs in a single dashboard page. |
| FR11 | The system can export existing diagrams and data into PDF documents. |

3.2. Component Analysis
In order to meet the requirements, several components are used to build the application. Mapping between the requirements and the components used can be seen in Table 1.

Table 2. The List of Mapping Between Requirements and Components

| Requirement | Component Name |
|-------------|----------------|
| FR1         | Tweetsharp     |
| FR2         | LibSVM         |
| FR3         | LibSVM         |
| FR4         | Not Use Component |
| FR5         | Kendo UI       |
| FR6         | Kendo UI       |
| FR7         | Kendo UI       |
| FR8         | JQCloud        |
| FR9         | Kendo UI       |
| FR10        | Metronic       |
| FR11        | Kendo UI       |

From Table 2, it can be seen that there are 5 external components used, which are Tweetsharp, LibSVM, Metronic, Kendo UI and JQCloud. Tweetsharp Library was chosen because this library is recommended by professional IT[16]. LibSVM was chosen because libSVM is the most widely used library for software development using SVM, which has a good performance [17]. This research uses Metronic in order to easily adapt with the existing application of PT KCI. To display graphs in the dashboard, Kendo UI is used because it is easy to use and Open Source. This research also uses JQCloud library, because the library is open source and easy to use.

3.3. Requirement Modification
In the requirement modification stage, the functions to be used are determined from the components obtained in the Component Analysis stage. Each component has a function that is used to meet one of the system requirements.
a. Tweetsharp

The functions of Tweetsharp library, which will be used in this research, are OAuth and search function. OAuth or Open Authorization function is used to gain access to Twitter so the apps can use Twitter Rest API. Oauth on Twitter is required because the application built in this research is Third Party Application, that needs Token Security to get access to Twitter Rest API. Token Security is obtained by creating a virtual application on dev.twitter.com. The second function, Search function, is used to search tweets that are related to specific keyword, that is "CommuterLine". In the Tweetsharp library, modifications are needed because the system has to get all tweets on the D-1 day.

b. LibSVM

In this library, the function used is the linear kernel SVM classification function. The training data format used by LibSVM is a sentence that has been in the form of basic words and has been through the preprocessing stage.

c. Kendo UI

The function of Kendo UI used are a function to display graphs or diagrams and export them to PDF files. The graphs used are Donut Charts, Bar Charts, and Time Series Charts.

d. Metronic

The Metronic component used is the admin template for using as the dashboard display.

e. JQCloud

The function used from JDCloud in order to display a wordcloud graph is Wordcloud Template.

3.4. System Design with Reuse

The design of the system to be built in this study can be seen in Figure 2. There are two modules built, namely Backend module and Dashboard module.

1. Backend Module

The Backend module was set up to run automatically by Windows Task Scheduler every day at 00.00. The components and processes in the Backend module are:

a. Tweetsharp

Tweetsharp is used in the Backend module to make it easier to crawling data from twitter. Tweetsharp is used for OAuth Twitter using API key and Token key to search all tweets that associated with the keyword "commuterline". All the twitter gathered from this process are saved in the database. This process will be ran automatically everyday.

b. Preprocessing

Figure 2. System Design with Reuse
All the data should be retrieved from the database before running preprocessing. There are some steps in this process which are:

1. Eliminating url (link): all url (link) in tweet must be removed.
2. Eliminating mention: all username (mention) on tweet must be removed.
3. Eliminating RT: all RT (retweet) on tweet should be omitted.
4. Removing the hashtag: all hashtags in the tweet should be omitted.
5. Eliminating punctuation (delimiter): all punctuation (delimiter) in the tweet should be removed.
6. Eliminating numbers: All numbers on the tweet should be omitted.
7. Case Folding: all text in the tweet should be in lower case.
8. Eliminating excessive spaces (white space): all white space in the tweet should be removed.
9. Removing stop words: all stop words in the tweet must be removed.
10. Normalization of abnormal words: all words must be changed into normal word form and there is no abbreviation.
11. Stemming and formatting to lower case: all words must be converted into basic word form and all letters should be changed into lowercase.

c. LibSVM Library

LibSVM library requires training data containing sentences that have been through the preprocessing process and class labeling, which was done manually. This research provided two training data, that are data of sentiment training, and data of complaint training. LibSVM creates models from sentiment training data, then the tweets are classified using the created model.

The sentiment training data were classified into three types of sentiments, that are positive (class 1), negative (class -1) and neutral (class 0). The tweets data that have been classified as negative sentiments are updated to the database in the "complaint" column and they will be the complaint training data.

The training data of the complaints classification are divided into 6 classes using libSVM, which are class -3 as Queue and Delay category, class -2 as KRL Capacity category, class -1 as Hygiene category, class 1 as Facility category, class 2 as a Service category and class 3 as the Security category. The training process will create a new model of complaints classification, that are used to classify tweets. The training process does not change every day, but it will be imported to the application anytime it has been changed. Imported model can lighten the application work and shorten the processing time.

d. Line Classification

This module classify tweets based on the route (line) of the Commuterline, by detecting the existence of a transit station in the tweet. If there is found a same word with the existing keyword in the transit station list, it will check again whether there is another station word. If it does not present, then it will be inserted into the line as defined in the transit station table. As an example, there is a tweet: "Train Manggarai-Bekasi late", then the application will detect the word "Manggarai" as a transit station and it will check again if there is another station word. Because it found the word "Bekasi", then the tweet will be updated as the Bekasi Line.

e. Worcloud Tokenization

Wordcloud Tokenization is a function to calculate every word that appears in all tweets from the customer to be displayed on the wordcloud graphic to see what words appear most frequent. Wordcloud displays data tweets that are split into 1 word and 2 words.

2. Dashboard

After the Analysis application is completed, the next step is to build the Dashboard module. This module is used to display the results of the generator module in the form of graph, table and word cloud. Figure 3 shows the dashboard mockup to be created. On the top left side there is a PT logo. KCI. On the top right side there is a button to logout. In the dashboard page, there are 6 features that are shown, which are Complaint Type Percentage chart, Top 5 Most Active User graph, Top 5 Complaint Tweet graph, Complaint Worcloud, Sentiment Per Day graph and Sentiment Per Month graph.
3.5. Development and Integration

The development is divided into two stages, namely development of Backend module and development of the Dashboard. Development was done by using Visual Studio 2015 Community Edition and SQL Server 2014 for database.

1. Backend Module

   The Analytics app was built using the .NET framework. The components used in the analytics application are Tweetsharp and LibSVM. The Analysis app will be ran automatically at every 00.00 server hours using Windows Task Scheduller

2. Dashboard Module

   The Dashboard built using Metronic components and Kendo UI to display the analysis results in the form of graphs. This module is web based application which was built using the .NET framework. There are six graphs shown in the dashboard, namely:

   a. Type of Complaints Donut Chart
Figure 4 shows a complaints graph. The data is displayed in the form of donut charts, with two date pickers to receive date input range from the data you want to display. This default graph displays d-1 data. Figure 4 describes the percentage comparison of the number of tweet customer data over a given time range based on the complaint categories.

In the process, complaints are classified using LibSVM with its own training data for the classification of complaints types. LibSVM retrieves training data stored in the database. Then classify one by one tweet into the appropriate complaint category. From the graph we can see what categories of complaints appear most. So that PT KCI can know the most complained by the customer at that time and immediately take action to overcome the problems that occur and prioritize responding to complaints from customers associated with the most complaints.

b. Top 5 Most Active Customer

Figure 5 shows the Top 5 Most Active Customer Graph. The data shown are the most active 5 posters posting the negative tweets associated with the Commuter Line account. If we highlight the bar part of the user, the tooltip will display the number of tweets from the Customer. This feature is intended to monitor whether a customer is a spammer who intends to give commuterline bad image or not. The dashboard user can select the date range for the data that want to be displayed. Default This graph displays the d-1 data.

c. Top 5 Complaint Tweet

Figure 6 shows Top 5 Complaint Tweet table. The data displayed is Top 5 Tweets from the most popular customers based on the most retweet count. The data displayed are the text of the tweet, the user who tweeted, and the date of tweet. From picture 6 we can see which tweets are popular and many directweet by other customers. The user nameboard can select the date range for the data that want to be displayed, the default is to display the d-1 data.

d. Wordcloud

Figure 7 Displays the word cloud. The word cloud contains words that often appear on tweets so that the user can find out the general description of the complaint tweet keluhanardi customer. From Figure 7 we can see what topics are being discussed by many customers. Users can select the date range for the data they want to display. The default graph in Figure 7 shows the h-1 data. The word in the show is 1 word and 2 words that most appear. Words are taken from the database table, ie data that has been done preprocessing, so the word that appears is the word base. For 1 word data taken from the data in the database separated by space. As for the 2 word data, the previous word is stored in the temporary variable, then added with the next word.

e. Sentiments Per Day Graph

Figure 8 shows the time series of the number of tweets from the customer per day. Users can select the date range for the data they want to display. As the default, it display the data from D-7 day to D-1 day.

f. Sentiment Per Month Graph
The sentiments per month graph is shown in Figure 9. It shows the time series of the number of tweets from the customer per month. From the graph we can develop the number of tweets from the customer whether increased or decreased in the range of months. Users can select the date range for the data they want to display. As the default, it shows the last 3 months data.

3.6. System Validation

In the System Validation stage testing of the applications built in this study. However, prior to testing, training data is required for SVM classification. The data collected is a tweet that goes into @CommuterLine Twitter account, then do preprocessing. Preprocessing produces data in the form of a tweet in the form of a basic word sentence. After preprocessing then done manual classification to determine the tweet into the class negative, positive or neutral. Once completed, the training data is stored in the database.

System testing is done in two stages, namely functional and non-functional. Functional system testing is done to determine whether the system built meet the needs obtained in the requirement analysis stage. Testing was done by installing the application in the server of PT. KCI, then it was tested by the user. The result was all the system functional requirements match with the functional requirements. Non-functional system testing is performed to measure performance. Performance measured is classification accuracy with SVM and processing time for analytics generator applications. The test results show the average of 80.4% sentence classification accuracy, and the accuracy of the classification of 82.3% complaint types from 3167 data. Then performed performance measurement based on application process generator time. From 3 times the measurement using stopwatch on December 18, 19 and 20 got the average process time 3 minutes 25 seconds. The data testing used is crawling data on 14-19 December 2017.

4. Conclusion

This research has successfully built a system for analysis of critical data from Twitter PT. KCI uses the Support Vector Machine method. The system is built using Component-based Software Engineering method. In this system there are two applications namely Application Generator and Dashboard. The generator app works to crawling data from twitter, preprocessing, classifying using LibSVM, and creating wordcloud. In the dashboard there are 6 features, which are Complaints graph, Top 5 Most Active Customers, Top 5 Negative Tweet, Wordcloud, Sentiment Per Day Graph, Sentiment Per Month Graph.

The graphs on the built dashboard meets the requirements of the user. The result of sentiment classification using SVM method performed on Generator Application showed 80.4% as accuracy of classification and 82.3% complaint classification accuracy from 3167 with average processing time 3
minutes 25 seconds. Error classification tweet in testing that has been done generally due to limited training data.

References
[1] PT Kereta Commuter Indonesia, “Sekilas PT Kereta Commuter Indonesia,” http://www.krl.co.id, 2019. [Online]. Available: http://www.krl.co.id.
[2] F. Ghofrani, Q. He, R. M. P. Goverde, and X. Liu, “Recent applications of big data analytics in railway transportation systems: A survey,” Transp. Res. Part C, vol. 90, no. January, pp. 226–246, 2018.
[3] E. Mogaji and I. Erkan, “Insight into consumer experience on UK train transportation services,” Travel Behav. Soc., vol. 14, no. April 2018, pp. 21–33, 2019.
[4] S. Gitto and P. Mancuso, “Improving airport services using sentiment analysis of the websites,” Tour. Manag. Perspect., vol. 22, pp. 132–136, 2017.
[5] D. Pham and A. Le, “International Journal of Approximate Reasoning Exploiting multiple word embeddings and one-hot character vectors for aspect-based sentiment analysis,” Int. J. Approx. Reason., vol. 103, pp. 1–10, 2018.
[6] D. M. D*, S. C*, and A. Ganesh, “Sentiment Analysis: A Comparative Study On Different Approaches,” Procedia - Procedia Comput. Sci., vol. 87, pp. 44–49, 2016.
[7] B. Ghaddar and J. Naoum-sawaya, “High dimensional data classification and feature selection using support vector machines,” Eur. J. Oper. Res., vol. 265, no. 3, pp. 993–1004, 2018.
[8] N. Sulistyawan and S. W. Sihwi, “Mapping Criminal Location Entity from Indonesian Online Newspapers,” vol. 7, pp. 112–117, 2018.
[9] I. F. R. Roji, S. W. Sihwi, and A. Doewes, “Development of Regional Public Election Analytics Application In Indonesia Using Data from Twitter,” in 2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2018, pp. 394–399.
[10] S. Anastasia and I. Budi, “Twitter sentiment analysis of online transportation service providers,” in 2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS), 2016, pp. 359–365.
[11] D. Dowding et al., “Dashboards for improving patient care: Review of the literature,” Int. J. Med. Inform., vol. 84, pp. 87–100, 2014.
[12] D. Li, J. Cao, and Y. Yao, “Big data in smart cities,” Sci. China Inf. Sci., vol. 58, no. October, pp. 1–12, 2015.
[13] F. H. Khan, U. Qamar, and S. Bashir, “eSAP: A decision support framework for enhanced sentiment analysis and polarity classification,” Inf. Sci. (Ny), vol. 367–368, pp. 862–873, 2016.
[14] F. Gutiérrez, K. Seipp, X. Ochoa, K. Chiluiza, T. De Laet, and K. Verbert, “Computers in Human Behavior LADA: A learning analytics dashboard for academic advising,” Comput. Human Behav., no. December, pp. 0–1, 2018.
[15] I. Sommerville, Software Engineering, 9th edition. Addison Wesley, 2011.
[16] D. Esposito, “How Use Tweets Sharp Push Twitter Updates Application,” 2012.
[17] C.-C. Chang and C.-J. Lin, “LIBSVM: A Library for Support Vector Machines,” ACM Trans. Intell. Syst. Technol., vol. 2, no. 3, p. 27:1–27:27, May 2011.