Leveraging the Text Mining to Automate the Customer Helpdesk Systems

Paramesh S.P.
Department of CS & E
U.B.D.T College of Engineering
Davanagere, Karnataka

Shreedhara K.S.
Department of CS & E
U.B.D.T College of Engineering
Davanagere, Karnataka

ABSTRACT
Customer helpdesk system plays an important role in assisting the end users or customers of the organization to get the resolutions for their service-related problems. In a typical customer helpdesk service environment, manual classification of tickets may involve misclassification and hence results in addressing the ticket to a wrong domain expert group. There is a need to develop an automated ticket classifier system which does the auto categorization of helpdesk tickets. This research paper presents such an automated helpdesk ticket classifier by using the artificial intelligence concepts like text document classification and natural language processing techniques. The proposed helpdesk ticket classifier model categorizes the incoming ticket by mining the unstructured text description entered by the end user. The research work uses the vector space model with TF-IDF term weighting approach for the representation of helpdesk tickets and Chi-square term selection technique for the dimensionality reduction. Finally, the classification techniques like linear Support vector machines, ID3 Decision trees and ensemble Random Forest are used to build an automated ticket classifier model. Real world helpdesk ticket datasets belonging to two different domain areas are used for the experimental purposes. The effectiveness of the chosen ticket classifier models is measured using various model evaluation metrics. Ensemble based Random Forest classifier performed well when compared to all other considered models. Automated ticket classifier systems result in faster ticket resolution, effective resource utilization and enhanced growth in business.

General Terms
Machine Learning, Natural Language Processing, IT service management systems

Keywords
Text mining, Helpdesk systems, Ticket classifier, Feature selection, Support vector machines, Random Forest

1. INTRODUCTION
Customer Help desk systems play an important role in assisting the users of the organization to get the resolutions to their problems regarding the organizational services. Help desk systems acts as a single point of contact (SPOC) for all types of problems experienced by the users and through which users can log the problem tickets and obtain the solution to their business issues [1].The users can be end users, customers, employees of the organization etc. Nowadays, service desk systems are used in almost all business verticals such as telecom, retail, banking and finance, healthcare, manufacturing, Information Technology (IT), education etc.

In a typical web-based helpdesk systems, the users create the problem tickets by manually selecting an appropriate ticket category from the dropdown list and choosing the fields like sub category, severity, priority and other required fields if any and then submits the ticket. The user also enters a brief summary about the ticket in the free form description field. Once the user submits the ticket after entering the ticket details, depending upon the issue category selected, the ticket will be addressed to the designated domain team for the resolution. Further, the expert team will analyze the tickets and responds with proper resolution within due time depending upon the priority of the tickets [2]. In some helpdesk systems like call based or email-based systems, there will be a dedicated dispatcher or helpdesk agent who does the initial analysis of the problem tickets at the first level and then dispatches the ticket to the next higher levels requesting for the resolution.The basic workflow management in a customer helpdesk system is depicted in the following Fig.1.

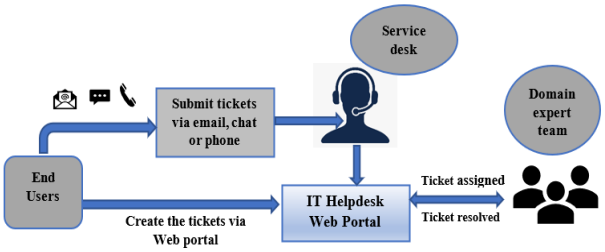

Fig. 1: Basic workflow management in a typical helpdesk system

Irrespective of the type of the helpdesk systems, all these systems involve manual categorization of tickets either by end users or by the first level helpdesk agent. Manual selection of tickets may result in wrong categorization due to lack of domain knowledge about the problem categories and moreover it is a time-consuming process [3]. The end users and the service desk agents should have a proper domain knowledge in order to select the correct problem category while submitting the tickets. Wrong selection of issue category in turn ends up in forwarding the ticket to the wrong resolution group and hence causes ticket reassignment and resolution delay. Further it also results in customer satisfaction deterioration and unnecessary resource utilization. In general, if there are many such misclassification of problem tickets in the service management systems then it impacts the business at the end of the day and affects the business growth.

To address all these issues concerning manual categorization of helpdesk tickets, automated ticket classifiers can be developed to auto categorize the incoming helpdesk tickets using AI based machine learning approaches [4,5,6]. In this research work, techniques like text mining and basic natural language processing are leveraged to build such an intelligent

automated system. The proposed methodology builds the ticket classifier system using historical ticket data containing summary of the tickets and the corresponding ticket label. Since the ticket description and its related categories are used to train the proposed ticket classification system, various other attributes of training data have only a limited role in building the proposed model. The users ticketsummary is composed of natural language text and is highly unstructured in nature [7,8]. The proposed model parses such an unstructured description, analyses it and classifies the incoming ticket into a predefined category.

In the proposed research work, automated helpdesk ticket classifier is modelled using text classification approaches namely support vector machines (SVM), decision trees (DT) and ensemble based Random Forest classifier. Real helpdesk ticket datasets belonging to two different domains areas namely IT infrastructure and Banking were used for experimental purpose. The proposed model auto categorizes help desk tickets into pre-defined category so that the tickets will be dispatched to the associated expert group.

The advantages of the proposed automated ticket classifier compared to traditional helpdesk systems includes simple user interface, faster ticket resolution, efficient use of organizational resources, improved end user satisfaction and growth in business.

2. LITERATURE REVIEW

Since this research work mainly focuses on building the automated helpdesk ticket classifier systems based on text classification approach, some of the existing literature in the area of automation of helpdesk systems to categorize the problem tickets is presented first followed by some of the studies in the field of text mining.

Feras Al-Hawari et al. [2] proposed a methodology for accurate ticket classification in IT helpdesk system using machine learning techniques. The method uses Support vector machines (SVM) to classify the help desk tickets. To manage and report the issue tickets, the user interface of the proposed system provides an efficient administrator and user view functionalities.

The work in [3] uses a rule-based machine learning approach to classify the problem tickets and the proposed rule-based classifier performed well in comparison with traditional naive bayes model when insufficient labelled data exists.

Paramesh S.P et al. in [4] investigated the effectiveness of various supervised machine learning approaches to segregate the IT infrastructure service desk problem tickets. Naive Bayes, SVM, Logistic regression and K-Nearest Neighbor models are employed to design the service desk ticket classifier, SVM reported good accuracy compared to other chosen approaches.

Silva et al. [5] proposed a methodology to automatize the incident classification procedure using SVM as the baseline model. The performance of the incident classifier is validated using a real-time incident ticket data and experiments findings shows that SVM model outperformed other chosen classifiers.

Roy S et al. [6] proposed a non-negative matrix factorization (NMF) based approach to extract the topics and clusters from the unstructured summary of the IT infrastructure tickets. The clusters thus formed are automatically labelled by extracting the semantic labels from each topic. The quality of the topics and clusters are evaluated using the Silhouette index and Davies-Boudin index metrics. The efficacy of the proposed unsupervised NMF approach is compared with other topic modelling techniques like Latent Dirichlet Allocation (LDA) and the proposed method performed well.

Agarwal et al. [7] proposed a cognitive IT support system that identifies the issue category, investigate the cause of the issue and provides automated ticket resolution. The system leverages the already closed tickets containing ticket summary, its category and resolution as provided by domain experts while closing the tickets to build such a cognitive system.

Dasgupta et al. [8] proposed an automated model called BlueFin to identify the problem from the users unstructured ticket descriptions. The model leverages two-step analysis for the problem identification which involves correlation of different data sources with the tickets to get quality dataset thereafter context-based classification using the correlated dataset. The model efficacy is compared with SmartDispatch an SVM based ticket classifier, Bluefin tool outperformed SmartDispatch for most of the samples.

Mucahit et al. [9] developed a ticket classifier model using various supervised ML algorithms. The proposed method uses the problem definition, category, sub category and ticket descriptions to build the classifier model. Experiments are conducted using a dataset consisting of approximately ten thousand issue tickets collected from ITU Issue Tracking System. Results shows that model performance relies on the dataset, term weighting approach and machine learning algorithms used to build the model.

Sebastiani in [10] discusses in detail the applications of text categorization in various domains, machine learning approaches to implement the text categorization, document representation schemes, dimensionality reduction techniques using term selection and extraction methods, various classifier construction and evaluation measures.

A brief survey of different text representation schemes, machine learning and hybrid approaches used for text document classification tasks are covered by A.khan et al.[11]. Authors also does the comparative study of various techniques used for text classification.

Joachims in [12] discusses the use of Support vector machines for text classification problems. Theexperimental findings shows that SVM's works well with high dimensional and linearly separable data like text and SVM outperformed other conventional classification algorithms in several text classification tasks. The various kernel functions used with SVM to solve the non-linearly separable cases in text classification is discussed in [13].

Kowsari et al. [14] provides a brief overview of techniques used in various phases of the text classification pipeline. Basic text pre-processing methods, text representation and feature dimension reduction techniques, different text classification and model evaluation metrics are discussed. Real word applications of text classification along with the limitations of various machine learning techniques used in text classification are also discussed.

3. PROPOSED METHODOLOGY

The design and implementation of the proposed help desk ticket classification system is based on the classical text document classification process and hence all the required phases of a typical text classification are involved in implementing this work. The design of the proposed helpdesk ticket classifier is depicted in the Fig.2.

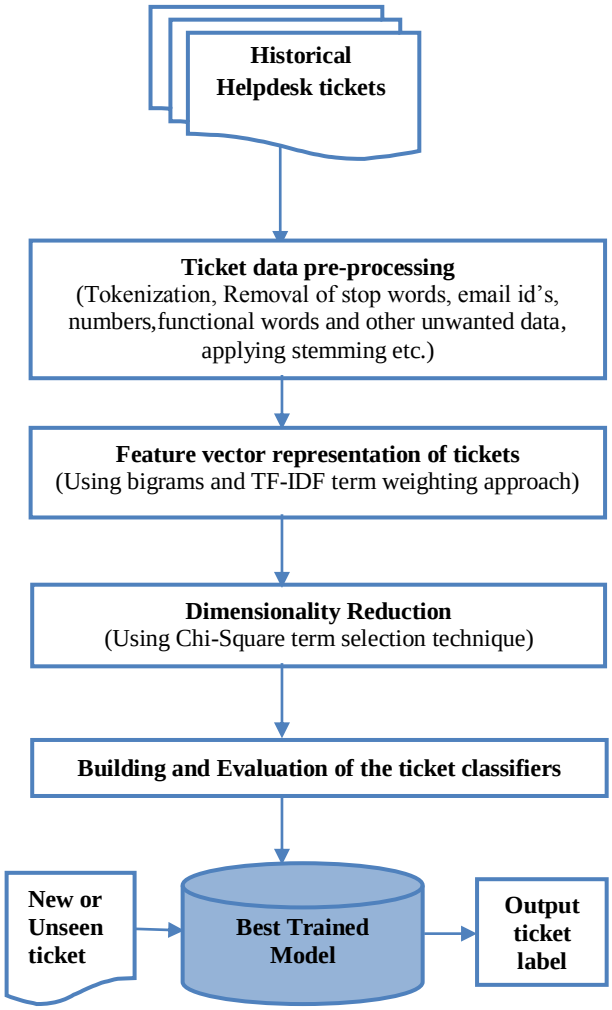

Fig.2: Design of the proposed helpdesk ticket classifier

Each of the components involved in the design of the proposed helpdesk ticket classification model are explained as follows.

3.1 Ticket data collection

Historical helpdesk ticket dataset consisting of unstructured ticket summaries and associated ticket categories are used for training the proposed model. The initial dataset may contain other fields like priority, severity, submitter name, attachments etc. but user ticket description and corresponding label plays an important role in building the proposed ticket classifier system. In this research work, real historical datasets belonging to two different domains namely IT infrastructure and Banking containing multiple classes are used for experimental purpose.

3.2 Data pre-processing

Exploratory data analysis of the original training data of both chosen datasets revealed that there exist data imbalance problems in the datasets. Hence, random resampling based on oversampling and under sampling approaches are leveraged to fix the imbalance issue. Further analysis of the training datasets revealed that there exists lot of unwanted and irrelevant features in the datasets. Accurate and efficient ticket classifiers can be developed by using most representative features present in the dataset. Hence, there is a necessary to extract and eliminate these unwanted attributes from the dataset and to retain only those important features that contributes in classification task.

Data cleansing completely depends on the dataset chosen for the research work. The datasets considered for this research work had lot of noisy and irrelevant features like stop words, email ids, numbers, names, special characters, date and time, phone numbers, functional words etc. In this work, the process of data cleaning of the training dataset involves lowercasing and tokenization of each ticket descriptions. Once the text is tokenized, various unwanted features are identified and filtered out from the dataset using the following approaches.

- A stop word is a frequently used English word like 'the', 'and', 'this', 'not', 'is' etc. and these words exists in the training data do not aid in classification task and moreover, these features consume more space and time. So, these stop words are filtered out from the training data with the help of standard English stop word list.
- Appropriate regular expressions are developed to remove the entities like email ids, special characters, date and time, phone numbers etc. found in the ticket descriptions.
- Part of Speech (PoS) tagging is performed on the dataset to remove the functional words such as pronouns, determiners, conjunctions etc. exists in the dataset.
- Named entities present in the training data are extracted and filtered out using the Named entity recognition (NER) concept.
- At last, stemming is applied on each feature of the training data to remove the suffix from a word and to reduce it to its base word.

Pre-processing of tickets results in reduced feature set and eliminates all the irrelevant features from the dataset that can be further used to model the efficient classifiers.

3.3 Representation and encoding of training data

A machine learning model doesn't understand the textual data and hence the pre-processed ticket dataset must be converted into a numerical representation before building the classifier models. In this work, a Vector space model or Bag of Words approach is used for representing the tickets. In this approach, each document i.e., ticket description in this case is represented by using feature vector representation where the n-gram features thus formed after the pre-processing of the dataset forms the attributes of the vectors. The feature vector is then encoded with Term Frequency-Inverse Document Frequency (TF-IDF) term weighting approach [9]. The $tf\text{-}idf$ of a term *t* in a given text document *d* is represented as below

$$tf\text{-}idf = tf(t,d) \times idf_t \quad (1)$$

Here, $tf(t,d)$ denotes the number of occurrences of the word *t* in a document *d*. The idf_t value is computed using the below equation

$$idf = \log\left(\frac{n_d}{n_d(t)}\right) \quad (2)$$

where, n_d is the total number of documents and $n_d(t)$ specifies the number of documents containing the term *t*.

3.4 Dimensionality reduction using term selection technique

Selection of most relevant feature set is an important aspect in

a text document classification problem that greatly improves the accuracy and efficiency of the target text classifier. Use of relevant feature set also avoids the over fitting of the training data. In this work, the dimensionality reduction is achieved using a term selection approach called chi-square (χ^2) metric to select the most representative attribute set [10,11]. Chi-square method generally measures the lack of independency between the term '*t*' and the category '*c*' in the dataset and is given below

$$\chi^2(t,c) = \frac{D(PS-RS)^2}{(P+R)(Q+S)(P+Q)(R+S)} \quad (3)$$

Here, *D* is the total number of documents i.e., ticket instance exists in the dataset. *P* and *Q* respectively represents the number of ticket instances having the term *t* in class *c* and in other classes of the dataset. *R* and *S* respectively represents the number of ticket instances do not containing the term *t* in class *c* and in other categories.After finding the chi-square ($\chi 2$) values of all the attributes of the pre-processed data, only those features with higher$\chi 2$ values are selected and used in classification.

3.5 Building helpdesk ticket classifier and evaluation of the model

Once the pre-processing of the training data is done followed by proper data representation and feature selection, the dataset would split into training and test set using 70:30 ratio with 70% of the data used for training the classifier model and remaining data to measure the effectiveness of the developed model. In the current work, helpdesk ticket classifier is built by using various classical methods like SVM, Decision trees and ensemble based Random Forest. Each of the classifier models considered in this research work are explained as below.

3.5.1 Support vector machines

Text classification problems generally have high dimension input space and are linear separable in nature. SVM's are largely used in automated text classification problem as they work well for high dimensional feature space [12]. SVM with linear kernel is chosen in this work as it is the most basic type of kernel and suits well for text classification problems. SVM's are generally used to implement binary classification task to predict whether a given input belongs to positive or negative category.

Consider a set of text documents as data points $x = [x_1, x_2, \ldots . x_n]$ where *n* is the number of data samples. Let c_i=1 andc_i=−1 respectively represents the positive and negative categories. Linear SVM algorithm involves finding the hyperplane $f(x) = 0$ that classifies the samples of the dataset and is mathematically given as follows

$$f(x) = w^T.x + b = \sum_{i=1}^{n} w_i . x_i + b = 0 \quad (4)$$

where, *w* represents the *n* dimensional weight vector and *b* is the bias.

The region between the two separating hyperplanes classifying positive and negative tuples is called "margin". SVM aims to maximize the region between two hyperplanes. The separating hyperplanes are represented as follows

$c_i f(x_i) = c_i(w^T.x_i + b) \geq 1$ for *i*=1, 2, ..,*n* (5)

An optimal hyperplane is one that creates the maximum margin between two hyperplanes separating the two classes. SVM model tries to find such an optimal hyperplane between two hyperplanes by solving the below optimization problem.

minimize $\frac{1}{2}|w^2|$(6)

with constraints$c_i(w^T.x_i + b) \geq 1$ for *i*=1, 2, ..,*n*(7)

Since this research work uses datasets containing multiple classes, target SVM ticket classifier model is generated by combining the predictions of multiple binary classifiers based on one versus all strategy [13].

3.5.2 Decision trees

Decision tree models are one of the important supervised machine learning techniques based on the inductive learning. Decision tree-based text document classifier represents a tree in which an internal node represents the term *t* of the document *d*, branches represent term weight *w* in *d* and leaf represents the class label *C* of the document. Decision trees canbe used categorize the given text document *d* to a particular category *C* by running through the query structure from root to a certain leaf node which represents the document category [14].

In this work, ID3(Iterative Dichotomiser 3) variant of decision tree model is used for building the ticket classifier. In ID3 at each stage, the algorithm calculates the information gain of each feature to select the best splitting attribute. Entropy of the dataset *S* is a measure of the amount of uncertainty or randomness in data and is mathematically represented as follows

$$h(S) = -\sum_{i=1}^{m} p_i \log_2 p_i \quad (8)$$

where,p_i represents the probability that an arbitrary instance in*S*belonging to category C_i and is determined as follows

$$p_i = \frac{|C_{i,S}|}{|S|} \quad (9)$$

where,$|C_{i,S}|$ is the total number of instances of a particular category*i*in the dataset *S* and |*S*| is the size of the dataset S. Information gain is a measure that specifies how well one attribute A_i classifies the training data and is determined for each attribute of the dataset using the following equation.

$$IG(S, A_i) = h(S) - \sum_{v \in values(A_i)} p(A_i = v) \times h(S_v) \quad (10)$$

In ID3 decision tree algorithm, the attribute with the highest information gain is selected at each stage to build the complete decision tree [15].

3.5.3 Random forest classifier

Random forest technique is an ensemble of decision tree learning method used for classification. It is an ensemble technique used to further improve the accuracy of the Bagged decision tree model. Random Forest classifier builds multiple decision trees models using the randomly selected subset of features of the training data [16]. Each individual model predictions are then averaged or aggregated to get the final model prediction. Random Forest considers subset of features to build the model in comparison to the bagged Decision tree that considers all the features at one time.

After training the helpdesk ticket classifier using different text classification approaches, the accuracy of the training models is found using the k-fold cross validation which gives the mean of the model scores. The effectiveness of the classifiers is also measured against the test set using accuracy and other evaluation measures like precision, recall and f-score. The ticket classifier which surpasses other chosen algorithms in performance is selected as the best model and is further used for predicting the class label of the new or unseen ticket instance.

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

Python 3.x version and its packages like pandas, matplotlib and sklearn are used to implement the proposed research work. Theexperimental results at various phases of the proposed helpdesk ticket classification are detailed as follows.

4.1 Data collection, analysis and pre-processing

In this research work, real world datasets from two different helpdesk systems belonging to different domains are used for experimental purposes. Dataset A and Dataset B used in this research work represents the helpdesk tickets related to IT infrastructure and Bank Consumer complaints respectively. The summary and the class label fields of the historical helpdesk datasets is used for training the model.Hardware problems, email related issues, OS issues, software installations issues, network problems etc. are some of the problem categories belonging to IT infrastructure dataset. The consumer complaints ticket dataset contains the categories related to loan, mortgage, bank account creation, debt collection etc. The exploratory analysis of the chosen ticket datasets revealed the following information as given in the Table1.

Table 1. Details of the datasets used

Description	Dataset A	Dataset B
Total number of ticket instances in the raw dataset	11200	66806
Number of ticket categories in the raw dataset	18	11

Analysis of the chosen helpdesk datasets also revealed that lot of unwanted and noisy data were present in the initial raw dataset. Data pre-processing is done to remove all the undesired data from the datasets using the techniques discussed in section 3.2 of this research work. The details about the features count before and after the data pre-processing task is given in Table2.

Table 2. Feature count details before and after the data pre-processing

Description	Dataset A	Dataset B
Total distinct words of the dataset prior to pre-processing of data	12240	64008
Number of unique features after filtering out the frequent words	9300	57110
Number of unique terms retained after eliminating all other unwanteddatalike email, functional words, special characters, numbers, date, time etc.	3923	36801

4.2 Ticket data representation and dimension reduction

Post pre-processing of the training data, each ticket descriptions of both the chosen datasets are represented as a numerical vector using the Bag of Words approach. Various n-grams like unigrams, bigrams and trigrams with TF-IDF term weighting approach is used in this workbut bigrams resulted in good accuracy compared to other n-grams while training the classifier.Relevant features are then extracted using Chi-square metric as a part of dimensionality reduction process. Experiments are also done using other term selection approaches like mutual information and information gain but the classifier using chi-square approach achieved good accuracy for the chosen datasets during the training phase.

4.3 Model building and evaluation

For each of the chosen datasets, out of the total number of ticket instances, 70% of randomly selected tickets are used in building ticket classifier models and the remaining 30% of the tickets are used for model validation. The performance of the ticket classifier model depends on the quality of the datasets and the feature set used for modelling. One cannot predict the best classification algorithm to be used for a given dataset at the initial stage. Popular text classification algorithms like SVM, decision trees and ensemble based random forest are used to build the proposed ticket classifier models. Accuracy performance of the chosen models on the training datasets is found using the k-fold cross validation technique with k=10. The average accuracy of various models using the of training data of Dataset A and Data set B is given in Table3 and illustrated in Fig.3.

Table 3. Accuracy performance of various ticket classifiers using k-fold cross validation

Classifier model	Dataset A	Dataset B
SVM	88.92	84.90
Decision tree	87.62	83.52
Random Forest	**90.65**	**86.21**

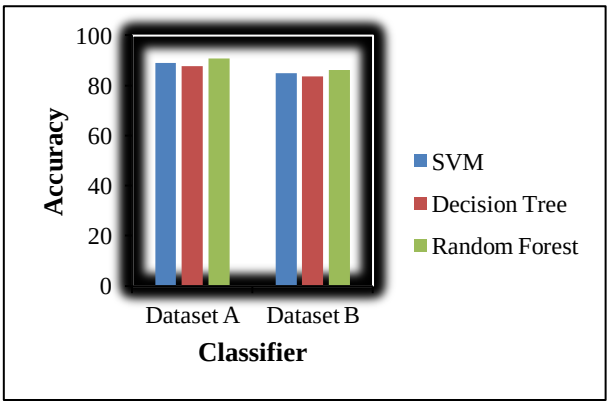

Fig. 3: Comparison of training accuracy of classifiers using k-fold cross validation

The results of the K-fold cross validation shown in Table3 and Fig.3 indicates that the Ensemble based Random Forest ticket classifier model having 90.65 % and 86.21% accuracy respectively for datasets A and dataset B performed well when compare to other alternatives. The performance of chosen classifier models is also evaluated on the test set (30%) of both the datasets using various evaluation metrics like precision, recall and f-score along with accuracy and the outcomes of the experiment are tabulated in Table4.

Table 4. Performance results of the ticket classifiers using the test data

Dataset	Metrics	SVM	Decision Trees	Random Forest
Dataset A	Accuracy	88.80	87.88	**90.90**
	Precision	88.70	87.21	**90.98**
	Recall	88.80	87.90	**90.90**
	F-score	88.60	86.42	**90.91**
Dataset B	Accuracy	84.50	83.78	**86.56**
	Precision	84.92	83.92	**86.43**
	Recall	84.61	83.10	**86.56**
	F-score	84.22	83.72	**86.56**

The performance results of various classifiers on the test data are also illustrated in Fig.4 and Fig.5 for the Dataset A and Dataset B respectively.

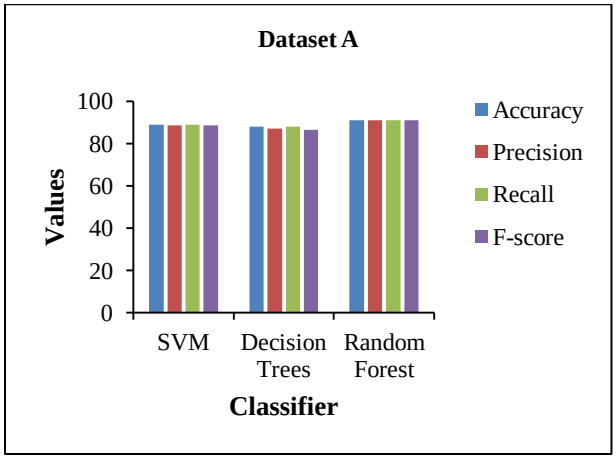

Fig.4: Comparative study of classifier performances using test set of Dataset A

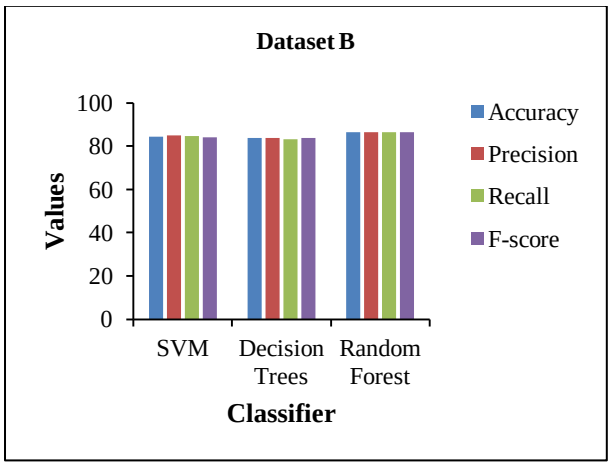

Fig.5: Comparative study of classifier performances using test set of Dataset B

The results shown in Table4, Fig.4 and Fig.5 illustrates that random forest classifier achieved a good accuracy of 90.90% and 86.56% respectively on the test instances of dataset A and dataset B. SVM achieved an accuracy of 88.8% and 84.5% on test set of dataset A and dataset B respectively. Similarly, Decision tree classifier model resulted in 87.88% and 83.78% accuracy against the test data of both the datasets. So, it is found from the prediction results of both training and test dataset that, ensemble based random forest classifier model outperformed other chosen classifiers. Hence, random forest is chosen as best performing model and is used to predict the ticket category of the new or unseen helpdesk tickets.

5. CONCLUSIONAND FUTURE WORK

In an IT service management systems like helpdesk systems, proper classification and routing of the user's problem tickets is a crucial step. Dispatching the helpdesk tickets to wrong domain expert group delays the resolution time, creates unessential ticket reassignments and resource utilization, decreases the end user satisfaction and affects the organization business. Current help desk systems may not handle the user's unstructured data efficiently. To overcome these issues associated with the traditional helpdesk systems, an automated helpdesk ticket classifier system is developed in this research work by leveraging the text mining and simple natural language processing techniques. The proposed model predicts the problem category of the helpdesk tickets automatically by mining the messy, unstructured natural language description of thetickets. Text classification algorithms likelinear SVM, ID3 based Decision trees and ensemble-based Random Forest classifier are used to build the proposed helpdesk ticket classifier. Real time helpdesk datasets belonging to IT infrastructure and Banking domains are used for experimental purposes. Effectiveness of the ticket classifier models is measured using various classifier assessment metrics. Experimental results shows that ensemble based random forest classifier achieved reasonably good accuracy compared to all other alternatives and worked well for both training and test sets of chosen datasets. The Random Forest ticket classifier achieved an accuracy of 90.9% and 86.56% on the test set of two chosen real datasets.

The proposed classifier results in sparse representation for larger datasets and requires manual extraction and selection of features using dimensionality reduction methods. So,in future this work can be further enhanced by exploring the dense representation schemes for text representation and automated techniques for feature selection and extraction.

6. REFERENCES

[1] P. Kubiak, S. Rass, "An Overview of Data-Driven Techniques for IT-Service-Management," IEEE Access, 6,63664–63688,2018.

[2] Al-hawari, Feras& Barham, Hala. A Machine Learning Based Help Desk System for IT Service Management. Journal of King Saud University.2019.

[3] Diao, Y., Jamjoom, H., & Loewenstern, D. Rule-based problem classification in it service management.In Cloud Computing, CLOUD'09,IEEE International Conference, pp. 221-228,2009.

[4] Paramesh S.P., Shreedhara K.S. Automated IT Service Desk Systems Using Machine Learning Techniques. In: International conference on Data Analytics and Learning, Springer LNNS,vol 43, pp.331-346, 2019.

[5] S. Silva, R. Pereira, and R. Ribeiro. Machine learning in incident categorization automation. In 2018 IEEE 13th Iberian Conference on Information Systems and Technologies (CISTI), pp 1-6, 2018.

[6] Roy S, Malladi VV, Gangwar A, Dharmaraj R. A NMF-based learning of topics and clusters for IT maintenance tickets aided by heuristic. In: Information systems in the big data era - CAiSE Forum, Tallinn, Estonia, Proceedings; p. 209–217, 2018.

[7] S. Agarwal, V. Aggarwal, A. R. Akula, G. B. Dasgupta and G. Sridhara, "Automatic problem extraction and analysis from unstructured text in IT tickets," in IBM Journal of Research and Development, vol. 61, no. 1, pp. 4:41-4:52, 2017.

[8] G. B. Dasgupta, T. K. Nayak, A. R. Akula, S. Agarwal, S. J. Nadgowda, "Towards auto-remediation in services delivery: Context-based classification of noisy and unstructured tickets", Proc. Int. Conf. Service-Orient. Comput. (SOC), pp. 478-485,2014.

[9] MucahitAltintas and CuneydTantug, "Machine Learning Based Ticket Classification in Issue Tracking Systems", Proceedings of International Conference on Artificial Intelligence and Computer Science, pp. 1-6, 2014.

[10] Sebastiani F., "Machine Learning in Automated Text Categorization", ACM Computing Surveys, vol. 34 (1), pp. 1-47, 2002.

[11] A. Khan, B. Baharudin, L.H. Lee, Kh. Khan, A Review of Machine Learning Algorithms for Text-Documents Classification, Journal of Advances in Information Technology, 1(1):4–20, 2010.

[12] Joachims, Thorsten. "Text categorization with support vector machines: learning with many relevant features." Paper presented at the meeting of the Proceedings of ECML-98, 10th European Conference on Machine Learning, Chemnitz, DE, 1998.

[13] Kaestner, Celso. (2013). Support Vector Machines and Kernel Functions for Text Processing. Revista de InformáticaTeórica e Aplicada. 20. 130. 10.22456/2175-2745.39702.

[14] K. Kowsari, K.J. Meimandi, M. Heidarysafa, S. Mendu, L.E. Barnes and D.E. Brown, "Text Classification Algorithms: A Survey", Proceedings of International Conference on Computation and Language, pp.1-7, 2019.

[15] Kotsiantis, S.B. Decision trees: A recent overview. Artif. Intell. Rev, 39, 261–283, 2013.

[16] Xu, Baojun& Huang, Joshua & Williams, Graham & Wang, Qiang& Ye, Yunming. Classifying Very High-Dimensional Data with Random Forests Built from Small Subspaces. International Journal of Data Warehousing and Mining, 8(2), 44-63, 2012.