Automated Ticketing System

Anthony Satuner, Avishek Burli, Geetha Charan Reddy

Abstract

One of the key activities of any IT function is to "Keep the lights on" to ensure there is no impact to the Business operations. IT leverages Incident Management process to achieve the above Objective. These Incidents are recorded in form of tickets and are assigned to certain support teams inorder to resolve them. However the time taken to Analyse, Assign and Re-Assign is really large and using ML and DL techniques this process could be automated, reducing FTE to focus on more critical issues. This report is an analysis perfomed on the following situation using a simulated dataset. An effective DL model was built which yielded an accuracy of ∼92 % accuracy on unseen data. It was also observed the issues faced in the dataset are majorly generated by job_scheduler error which can be solved by perfoming RCA (Root Cause Analysis) and automating Password-Reset process. Hence a cumulative Resource / FTE allocation reduction by approximately 37.34% is possible and Business can operate at ∼62.66% of original estimates.

Keywords: ML, DL, doc2vec, embeddings, LDA, FTE, SLA, Mojibake, Ticket Assignment, Incident Management Process, NLP

1 Introduction

The aim of this project is to build an Automated Ticket Assignment System using ML and DL so that the teams can focus on solving the issues much quicker. the effective assignment of the teams to the tickets is important to speed up incident resolution and reduce FTE. This report summarizes all the details of the project and analysis perfomed on the simulated dataset. The following sections will give more insight into the usecase of building a Automated Ticket Assignment System and also the Business Value of such a solution.

1.1 Real Problem

One of the key activities of any IT function is to "Keep the lights on" to ensure there is no impact to the Business operations. IT leverages Incident Management process to achieve the above Objective. An incident is something that is unplanned interruption to an IT service or reduction in the quality of an IT service that affects the Users and the Business. The main goal of Incident Management process is to provide a quick fix/workarounds or solutions that resolves the interruption and restores the service to its full capacity to ensure no business impact.

In most of the organizations, incidents are created by various Business and IT Users, End Users/ Vendors if they have access to ticketing systems, and from the integrated monitoring systems and tools. Assigning the incidents to the appropriate person or unit in the support team has critical importance to provide improved user satisfaction while ensuring better allocation of support resources. The assignment of incidents to appropriate IT groups is still a manual process in many of the IT organizations.

Manual assignment of incidents is time consuming and requires human efforts. There may be mistakes due to human errors and resource consumption is carried out ineffectively because of the misaddressing. On the other hand, manual assignment increases the response and resolution times which result in user satisfaction deterioration / poor customer service.

1.2 Business Value

In the support process, incoming incidents are analyzed and assessed by organization's support teams to fulfill the request. In many organizations, better allocation and effective usage of the valuable support resources will directly result in substantial cost savings.

Currently the incidents are created by various stakeholders (Business Users, IT Users and Monitoring Tools) within IT Service Management Tool and are assigned to Service Desk teams (L1 / L2 teams). This team will review the incidents for right ticket categorization, priorities and then carry out initial diagnosis to see if they can resolve. Around 54% of the incidents are resolved by L1 / L2 teams. Incase L1 / L2 is unable to resolve, they will then escalate / assign the tickets to Functional teams from Applications and Infrastructure (L3 teams). Some portions of incidents are directly assigned to L3 teams by either Monitoring tools or Callers / Requestors. L3 teams will carry out detailed diagnosis and resolve the incidents. Around 56% of incidents are resolved by Functional / L3 teams. Incase if vendor support is needed, they will reach out for their support towards incident closure.

L1 / L2 needs to spend time reviewing Standard Operating Procedures (SOPs) before assigning to Functional teams (Minimum 25-30% of incidents needs to be reviewed for SOPs before ticket assignment). 15 min is being spent for SOP review for each incident. Minimum of 1 FTE effort needed only for incident assignment to L3 teams.

During the process of incident assignments by L1 / L2 teams to functional groups, there were multiple instances of incidents getting assigned to wrong functional groups. Around 25% of Incidents are wrongly assigned to functional teams. Additional effort needed for Functional teams to re-assign to right functional groups. During this process, some of the incidents are in queue and not addressed timely resulting in poor customer service.

Guided by powerful AI techniques that can classify incidents to right functional groups can help organizations to reduce the resolving time of the issue and can focus on more productive tasks.

2 Dataset

The Dataset is provide by Great Learning from the course PGP-AIML as a part of the capstone project assignment. The Dataset Consists of the following columns.

- **Short Description**: Short description of the issue.
- **Description**: Detailed Description of the issue.
- **Caller**: Caller who has generated the ticket.
- **Assignment Group**: which group has to resolve the issue

Some Data Cleaning and EDA is perfomed on the dataset before any analysis can be done on the dataset.

3 Data Cleaning and EDA

It is observed there are lots of inconsistencies and issues with the current dataset. so it has to be further processed before perfoming any sort of analysis.

3.1 Data Cleaning

The Dataset is first checked for any inconsistencies and NULL treatement is applied. In the process it was identified that 8 records had missing data. The null were replaced with empty strings inorder to avoid any data loss. some encoding issues were identified in the Dataset. This issue is particularly refered to as **Mojibake**. it is the garbled text that is the result of text being decoded using an unintended character encoding. The result is a systematic replacement of symbols with completely unrelated ones, often from a different writing system.

The library ftfy (Fixes Text For You) has a greater ability to detect, fix and deal with such Mojibakes. It fixes Unicode that's broken in various ways. The goal of ftfy is to take in bad Unicode and output good Unicode. The grabled characters were successfully recovered using ftfy however it is observed that the row# 8471 is not English but Mandarine.

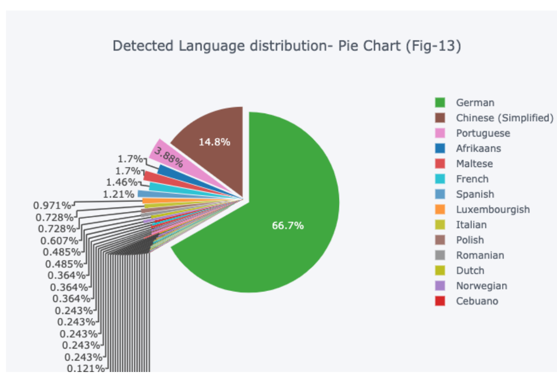

Fig. 1: Non-English tickets present in the Dataset

So the data in our hand is **multilingual** and it is quite not possible to derive embeddings for mix of multiple language which is clearly evident from **Figure 1**. Nearly 41 Languages were detected in the

Dataset. Its decided to translate the entire dataset into a single language of English. The GoSlate Library is used for this purpose to perfom any translation on the dataset and also identify the distibution of the issues.

other forms of cleansing functions applied are as follows:

- converting all letters to lower or upper case
- converting numbers into words or removing numbers
- removing punctuations, accent marks and other diacritics
- removing white spaces
- removing stop words, sparse terms, and particular words
- text canonicalization
- Lematization

The above process was achieved using a custom cleaning function and Spacy nlp pipelines.

3.2 Exploratory Data Analysis

It is observed that major number of tickets are places in **GRP_0** from **Figure 2**. the no of tickets assigned to GRP_0 is close to 4000 out of 8500 records, which is 46.8% in volume. Some resampling has to be applied inorder to balance it.

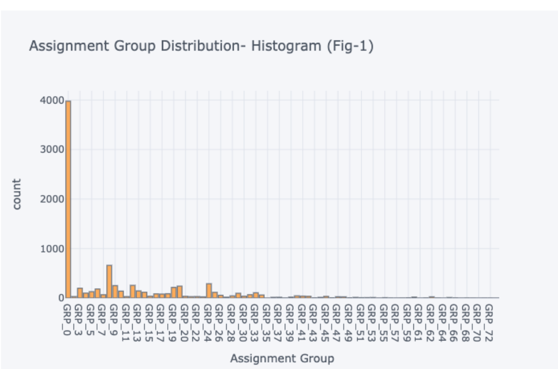

Fig. 2: Assignment Group Distibution

The 2^{nd} highest assignment group is GRP_8, which is just 7.78% of the total dataset and 1/6 th of the GRP_0.

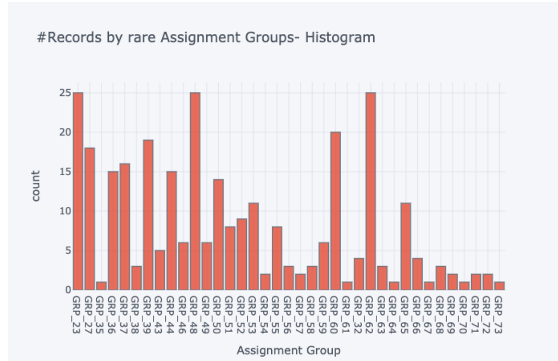

Fig. 3: Rare Tickets Distibution

There are 40 groups with just 25 or less tickets assigned in **Figure 3**, amongst which 6 groups happened to be assigned with just 1 ticket and 4 groups with just 2 tickets each. As assignment group attribue is the target column in our dataset, these tickets distribution shows the dataset is miserably imbalanced.

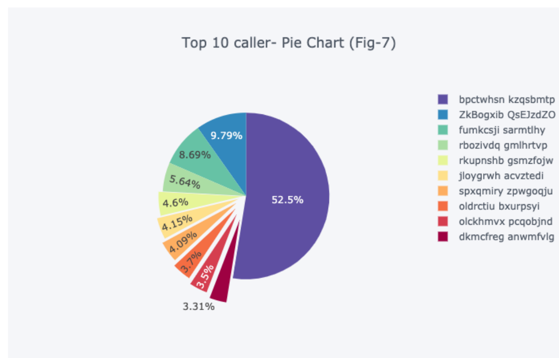

Fig. 4: Caller Contribution

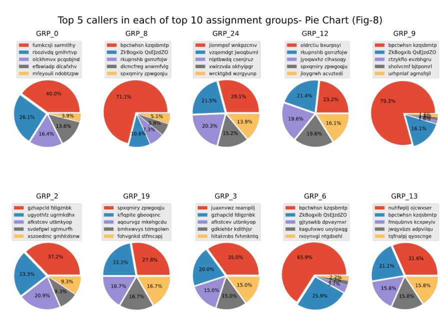

Fig. 5: Top 5 Callers in Top 10 Assignment Groups

The Callers Distibution is also imbalanced as only one particular caller is generating all the tickets **Figure 4**. From **Figure 5** its clear that for certain groups certain members are creating major tickets.

N-Gram analysis was also perfomed to understand any form or patterns present in the words of the dataset and It's indicative from **Figure 6** that the entire dataset speaks more about issues around

- password reset (1246 times)
- fail job_scheduler (1614 times)
- outlook (948 times)
- login (861 times)
- job fail (897 times)

Analysis on GRP_0 which is the most frequent group to assign a ticket to reveals that this group deals with mostly the maintenance problems such as password reset, account lock, login issue, ticket update etc. Maximum of the tickets from GRP_0 can be reduced by self correcting itself by putting automation scripts/mechanisms to help resolve these common maintenance issues. This will help in lowering the inflow of service tickets thereby saving the person/hour efforts spend and increasing the business revenue.

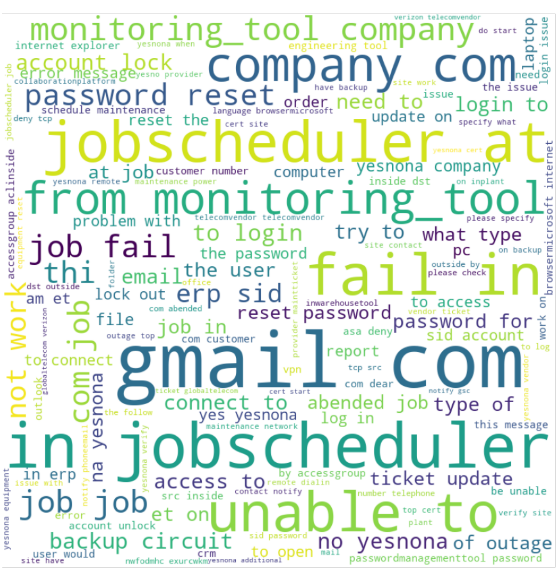

Fig. 6: Word Cloud of Summary Feature

After performing topic modeling using **LDA (Latent Dirichlet Allocation)** it was more clear how the allocation of the issues was to be done inorder to resample the GRP_0 tickets.

Fig. 7: Word Cloud of Summary Feature

The GRP_0 were now undersampled into the following categories for balancing the dataset. They are:

1. Communication Issue
2. Account/Password Reset
3. Access Issue
4. Other Issues

4 Feature Engineering

Some Notable Features generated as part of the Analysis are as follows:

- word length: # of words in a document.
- lines: # of lines in a document.
- shifts: time of the shift when the ticket was generated i.e general, night, morning.

All these features are merged into the Summary Feature so that its can be passed into the DL models with ease.

5 Pre-Processing

The dataset is first balanced using undersampling mentioned in the section 3 using LDA and oversampled. this will ensure that the models are more accurate on all classes and has sufficient training examples for the models.

for the word-embeddings gensim package was used to build a custom doc2vec model. this model provides the necessary embedding weights for training the DL models. for the Classical ML models TF-IDF (Term Frequency - Inverse Document Frequency) was used using sklearn package.

6 Modeling

6.1 Intial Training

Modeling was performed using many classical models and they can be refered from **Table 1**. Here its clear that the Classical Models are perfoming really well in accuracy . but on further inspection it is noted that the models are overfitting the dataset. so after careful analysis of the accuracy of the models, It is decided to use **Reccurrent Neural Networks with LSTM** as the Benchmark Model for further analysis.

Model	**Accuracy (%)**
Support Vector Machine - Linear	68
Deep Neural Network	64
Support Vector Machine - RBF	62
Random Forest	62
K Nearest Neighbours	61
Reccurent Neural Network with LSTM	57
Decision Tree	56
Recurrent Neural Network	54
Convolutional Neural Network	53
Multinomial Naïve Bayes Classifier	52
Recurrent Neural Network	5

Tab. 1: ML Models Results

6.2 Hyperparameter Tuning and Optimizations

Out of all the models tested, Support Vector Machine (SVM) under statistical ML algorithms and Neural Networks are performing better than all others. The models were highly overfitted and one of the obvious reason was the dataset was highly imbalanced. Ratio of GRP_0 to all others is 47:53 and there are 40 groups having less than or equal to 30 tickets assigned each.

	Accuracy (%)	
Model	**Train**	**Test**
Iteration 2 (increase LSTM units)	94.93	92.46
Iteration 3 (adding dense,dropout,batchnorm layers)	94.22	92.41
Iteration 4 (adding Adam)	94.15	92.88
Iteration 1 (changing dropout)	93.50	93.08
Base (Benchmark Model)	90.36	90.43

Tab. 2: DL Models Results

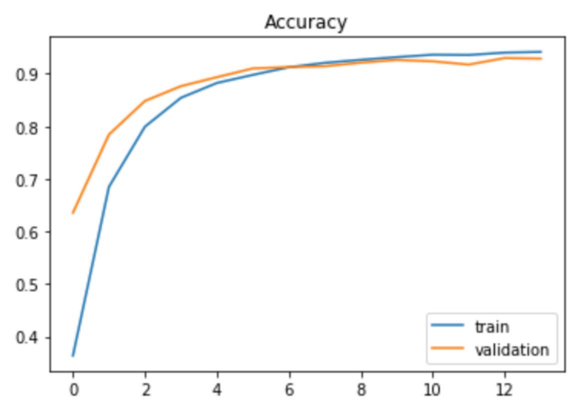

Fig. 8: Train vs Validation Accuracy

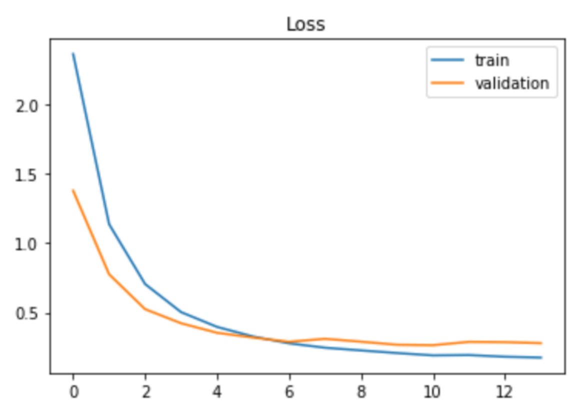

Fig. 9: Train vs Validation Loss

The accuracy of each flavors of LSTM model is as follows in the **Table 2**. This is clear indicative of how LSTM, in the family of RNN is efficient of dealing with textual data.

- There is a bump in the model performance upto the range 92.21 to 93.31 with 95% confidence level.
- Making the dataset balanced, helped the model to be trained more accurately.
- Creating custom word embbeddings helped finding better representation of keywords of the corpus.
- Hyperparameter tuning resulted in finding the model with more accuracy without overfitting, which is evident from the train vs. validation accuracy curve **Figure 8** and Loss Curve **Figure 9**.

7 Summary

Because of the Data Modeling its very clear on how the ticketing system model can classify the tickets effectively. there are other benefits of such system which are listed below.

Benefits:

1. Increase in Customer Satisfaction.
2. Decrease in the response and resolution time.
3. Eliminate human error in Ticket Assignment. (Which was ∼25% Incidents)
4. Avoid missing SLAs due to error in Ticket Assignment.
5. Eliminate any Financial penalty associated with missed SLAs.
6. Excellent Customer Service.
7. Reallocate (∼ 1 FTE) requirement for Productive Work.
8. Increase in morale of L1 / L2 Team.
9. Eradicate 15 mins Effort spent for SOP review (∼ 25-30% of Incidents or 531.25-637.5 Person Hours).
10. Decrease in associated Expense.
11. L1 / L2 Team can focus on resolving ∼ 54% of the incidents
12. Functional / L3 teams can focus on resolving ∼ 56% of incidents

∼1 FTE from L1 / L2 Team saved through automating Ticket Assignment can focus on Continuous Improvement activities. ∼ 25% of Incidents which is 2125 additional Incidents will now get resolved within SLA.

Additional Business Insights:

1. Root cause analysis (RCA) need to be performed on job_scheduler, to understand the cause of failure. No. of Incident Ticket reduction expected by performing RCA: 1928. 22.68% of Total Incident volume of 8500. Hence, we can reduce the Resource / FTE allocation also by approximately 22.68%.
2. Password Rest process need to be automated. No. of Incident Ticket reduction expected by automating password reset process:- 1246 14.66% of Total Incident volume of 8500. Hence, we can reduce the Resource / FTE allocation also by approximately 14.66%. Hence a cumulative reduction of 3174 Incidents means 37.34% reduction in Total Incident volume of 8500. Hence, cumulative Resource / FTE allocation reduction by approximately 37.34%. Business can operate at ∼62.66% of original Estimates.

References

[1] SuHan Han, "googletrans 2.4.0," 2018. https://pypi.org/project/googletrans/.

[2] DUONG Vu, "Generating wordclouds in python," 2019. https://www.datacamp.com/community/tutorials/wordcloud-python.

[3] Javed Shaikh, "Machine learning, nlp: Text classification using scikit-learn ,python and nltk," 2017. https://towardsdatascience.com/machine-learning-nlp-text-classification-using-scikit-learn-python-and-nltk-c52b92a7c73a.

[4] , "Googletrans api error - expecting value: line 1 column 1 (char 0)." https://stackoverflow.com/questions/49497391/googletrans-api-error-expecting-value-line-1-column-1-char-0.

[5] abo Samoor, "Multilingual text (nlp) processing toolkit," 2019. https://github.com/aboSamoor/polyglot.

[6] Mor Kapronczay, "Text preprocessing in different languages for natural language processing in python," 2019. https://medium.com/starschema-blog/text-preprocessing-in-different-languages-for-natural-language-processing-in-python-fb106f70b554.

[7] Olga Davydova, "Text preprocessing in python: Steps, tools, and examples," 2018. https://www.kdnuggets.com/2018/11/text-preprocessing-python.html.

[8] jacob peralta, "Text preprocessing in python — set – 1." https://www.geeksforgeeks.org/text-preprocessing-in-python-set-1/.

[9] MLWhiz, "Nlp learning series: Part 1 - text preprocessing methods for deep learning," 2019. https://mlwhiz.com/blog/2019/01/17/deeplearning_nlp_preprocess/.

[10] William Jaubert, "Eda - nlp - ml," 2018. https://www.kaggle.com/wil2210/eda-nlp-ml.

[11] Susan LI, "A complete exploratory data analysis and visualization for text data," 2019. https://towardsdatascience.com/a-complete-exploratory-data-analysis-and-visualization-for-text-data-29fb1b96fb6a.

[12] Pankaj Kishore, "It support ticket classification and deployment using machine learning and aws lambda," 2019. https://towardsdatascience.com/it-support-ticket-classification-and-deployment-using-machine-learning-and-aws-lambda-8ef8b82643b6.

[13] Rohit Ayinaparthy, "Text analytics using nlp(nltk) - it incidents," 2018. https://www.kaggle.com/rohitayinaparthy/text-analytics-using-nlp-nltk-it-incidents.

[14] Karol Zak, "support-tickets-classification." https://github.com/karolzak/support-tickets-classification.

[15] brandonmrose, "Document clustering with python." http://brandonrose.org/clustering.

[16] Shashank Kapadia, "Topic modeling in python: Latent dirichlet allocation (lda)," 2019. https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0.

[17] Shivam Bansal, "Beginners guide to topic modeling in python," 2016. https://www.analyticsvidhya.com/blog/2016/08/beginners-guide-to-topic-modeling-in-python/.

[18] Selva Prabhakaran, "Topic modeling with gensim (python)." https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/.

[19] , "kmeans text clustering." https://pythonprogramminglanguage.com/kmeans-text-clustering/.

[20] Mikhail Salnikov, "Text clustering with k-means and tf-idf," 2018. https://medium.com/@MSalnikov/text-clustering-with-k-means-and-tf-idf-f099bcf95183.

[21] SRK, "Simple feature engg notebook - spooky author," 2017. https://www.kaggle.com/sudalairajkumar/simple-feature-engg-notebook-spooky-author.