SELECTING THE BEST ML MODEL FOR YOUR IT SUPPORT FUNCTION

Automatic Ticket Assignment – Model Selection Report

Liu, Vanessa
vanessaliurs@gmail.com

Contents

Executive Summary

IT support is one of the most important engines to enable, maintain and monitor the daily business activities in an organization. In large organizations, IT support teams usually have many employees to ensure all the systems, applications, and other digital functionalities are working as expected. Whenever users encounter an issue/incident, they will raise a ticket through online portal for IT team to help resolve. As the volume of the tickets increases, ticket assignment requires more cognitive capabilities to ensure the efficiency and effectiveness.

This project is purposed to build an NLP (Natural Language Processing) algorithm to intelligently classify all the tickets into various categories, in order to decide to whom, or which team, each ticket is supposed to be routed to.

This report documents the approach (steps, logic, detailed insight) taking in the project followed by building the models and testing & tuning the performance. We will start from cleaning the data, handling exceptional values, as well as making sense of the data (basic EDA) to showcase the preprocessing outcome of the dataset and make sure the readiness of model building. After that, we will build different models to test the performance, and decide what is the most effective model.

Four types of models are tuned and compared: SVM, KNN, Bagging Classifier and Bi-LSTM.

The first 3 models use TF-IDF and Bi-LSTM uses word embedding. The reason of comparison is to find out which model performs the best in this business case. Final results turn out that Bi-LSTM usually performs better than other classification algorithms when we want our model to learn from long term dependencies; and SVM also produces good result in this case. The author still recommends Bi-LSTM model because in real-world practices, data will be more complex in a sequential way.

Business value of the solution is to help organizations improve IT support efficiency, saves long-term employment investment, and enhance user experience. It has high ROI in the sales cycle because it is a very parallel use case (replicable in many industries and companies) and easy to implement. Meanwhile, as RPA has been widely implemented, having this part of the process automated increases the chance to be integrated and achieve end-to-end automation.

More areas of improvement are encouraged given the constraint of the dataset. More entities are encouraged to be collected for more informative representation of real-world business. Additional information shall be added in a structured approach to enhance the accuracy. Making Short Description as a pre-defined dropdown list saves the implementation cost. Having country-specific portals could remove the differences between languages hence increase the accuracy.

From the perspective of implementation, a proper requirement gathering session is essential. The more digitized and structure the data is collected, the easier it will be to be integrated and implemented in a long term.

Problem Statement

As the business scale up and the number of employees grows, below concerns would occur in the process of IT ticket assignment:

- It is time-consuming to manually perform ticket allocation;
- It is costly to hire more workforces to perform this allocation and train them;
- There will be unavoidable mistakes in this allocation due to operational risk / lack of knowledge / subjective understanding, etc.;
- Bad employee user experience due to above three concerns.

Objective & Social Opportunities

What we are trying to do here is to understand the landscape of ticket allocation process and come up with an NLP-based algorithm to intelligently classify the tickets timely to the right team and person. The benefits include:

- Minimizes wasted time on manual allocation
- Saves cost to hire more workforces on manual allocation
- Minimizes mistakes caused by operational risk (because it will be automated) and subjective understanding / lack of knowledge (because the algorithm will be consistently applied)
- Enhances user experience by timely resolve their issues properly hence improves the business growth.

In the business world, it creates the opportunities because:

- All organization are seeking full automation in support functions. Basic data transportation could be done by RPA, but RPA does not have intelligence and cannot handle unstructured data. Embedding this algorithm would solve this problem by handling unstructured data with intelligence, hence integrated with RPA to achieve end-to-end automation.
- This is a highly replicable business case where we can parallelly seek business opportunities to sell to a wide variety of customers on the market. Basic tickets are all similar. We can sell it as a cloud-based AI fabric.

Solution & Approach

Below approach is adopted:

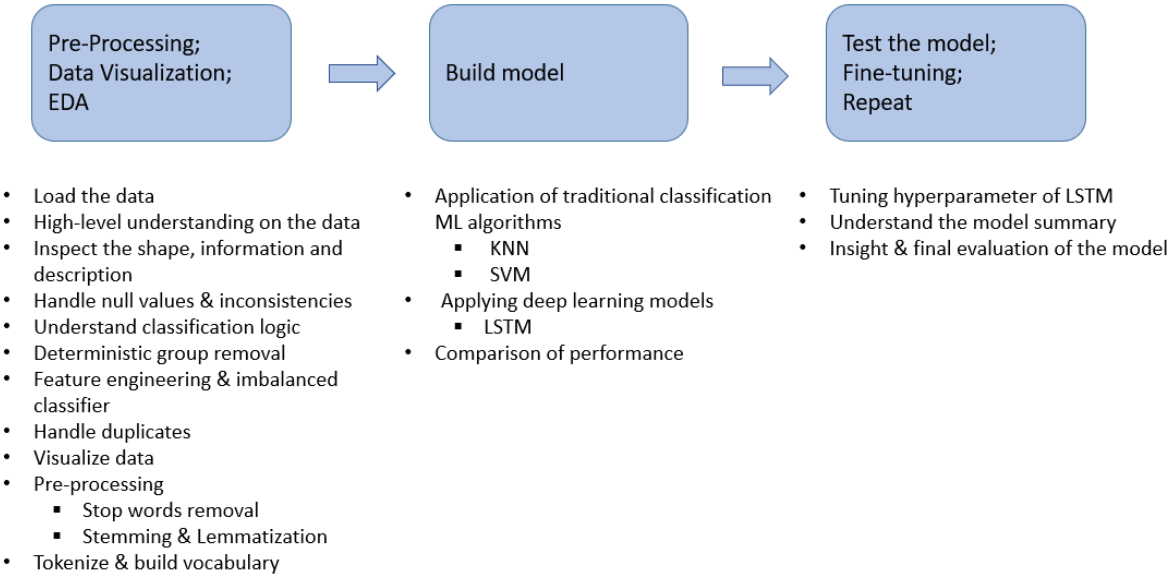

Figure 1: Project Approach

Solution Steps & Details

1. Load the data & high-level understanding

The dataset (8500, 4) has 8500 entities, and 4 columns: “Short description”, “Description”, “Caller”, “Assignment group”. “Assignment group” is the target parameter, and the other 3 are independent parameters. Each entity represents 1 ticket. Meanings of the columns are:

- Short description - a short summary of the whole issue. In some organizations, this data comes from a pre-defined dropdown list to narrow down the value range - not sure whether in this case, we will explore later. Some very simple and basic issues can be directly allocated according to short description without reading full description.
- Description - Full description from the raiser of the ticket. Further information and details are provided here to decide on the allocation.
- Caller - ticket raiser. From a very basic perspective, this column does not have a big impact on the allocation because they are simply the names of users. After a basic visualization, it is decided that this column can be avoided because it has no direct meaning or correlation to the classification.
- Assignment group - the target variable. Indicates to which group is the ticket allocated to. It is the final classification decision we need to predict.

We can basically understand the methodology of data collection according to above display. The tickets are raised on a real-time basis whenever the users encounter issue and decide to raise a ticket. Each user can raise as many tickets as they want to. This dataset is probably the historical data for the past month.

2. Inspect the shape, information and description

Distribution of the data is:

	Short description	Description	Caller	Assignment group
count	8492	8499	8500	8500
unique	7481	7817	2950	74
top	password reset	the	bpctwhsn kzqsbmtp	GRP_0
freq	38	56	810	3976

Figure 2: Distribution of Data

This means that:

- "Short description" and "Description" contain null values and the other two do not – need to be handled.
- A visualization of tickets raised by this caller "bpctwhsn kzqsbmtp" – they are not duplicates or system errors so we keep them.
- There are 74 groups in total to be classified – are they balanced? Do we need to keep all?
- Most issues are about password reset, and most tickets are raised by a specific user; most tickets (half) are assigned to GRP_0 – obviously imbalanced.

3. Handle null values & inconsistencies

First, null values must be handled properly after understanding the context. They are loaded:

	Short description	Description	Caller	Assignment group
2604	NaN	\r\n\r\nreceived from: ohdrnswl.rezuibdt@gmail.com\r\n\r\nhi,\r\n\r\n\r\n\r\nlink is not working. kindly resolve yhe issue on urgent basis.\r\n\r\n\r\n\r\nbest	ohdrnswl rezuibdt	GRP_34
3383	NaN	\r\n-connected to the user system using teamviewer.\r\n-help the user login to the portal.\r\n-issue resolved.	qftpazns fxpnytmk	GRP_0
3906	NaN	-user unable tologin to vpn.\r\n-connected to the user system using teamviewer.\r\n-help the user login to the company vpn using the vpn company vpn link.\r\n-issue resolved.	awpcmsey ctdiuqwe	GRP_0
3910	NaN	-user unable tologin to vpn.\r\n-connected to the user system using teamviewer.\r\n-help the user login to the company vpn using the vpn company vpn link.\r\n-issue resolved.	rhwsmefo tvphyura	GRP_0
3915	NaN	-user unable tologin to vpn.\r\n-connected to the user system using teamviewer.\r\n-help the user login to the company vpn using the vpn company vpn link.\r\n-issue resolved.	hxripljo efzounig	GRP_0
3921	NaN	-user unable tologin to vpn.\r\n-connected to the user system using teamviewer.\r\n-help the user login to the company vpn using the vpn company vpn link.\r\n-issue resolved.	cziadygo veiosxby	GRP_0
3924	NaN	name:wvqgbdhm fwchqjor\nlanguage:\nbrowser:microsoft internet explorer\nemail:wvqgbdhm.fwchqjor@gmail.com\ncustomer number:\ntelephone:-not available\nsummary:can't get into vpn - need to be on at 4:30 est and it wont' happen please help aerp!	wvqgbdhm fwchqjor	GRP_0
4341	NaN	\r\n\r\nreceived from: eqmuniov.ehxkcbgj@gmail.com\r\n\r\ngood morning,\r\n\r\nwhen trying to log on to erp i get this error below.\r\nplease help urgently as i can not process customer order.\r\n\r\n[cid:image001.png@01d20f2e.751db880]\r\n\r\n\r\n	eqmuniov ehxkcbgj	GRP_0
4395	i am locked out of skype	NaN	viyglzfo ajtfzpkb	GRP_0

Figure 3: Entities with Null Values

For this project, it is advised to remove these tickets with null values. Reason being they are string values, any replacement / refilling decided by me would be subjective and will impact the prediction. After dropping, the structure becomes (8491, 4).

In a long term, it is advised to improve the ticket portal - make both fields Short description & Description mandatory fields when users submit tickets.

4. Handle duplicates

We only consider it as a duplicate when all the 4 columns are the same because there is no standalone column for timestamp. After removal of duplicates, the dataset becomes (8408, 4).

5. Deterministic group removal

As we can see, most of the tickets are assigned to GRP_0. A better visualization:

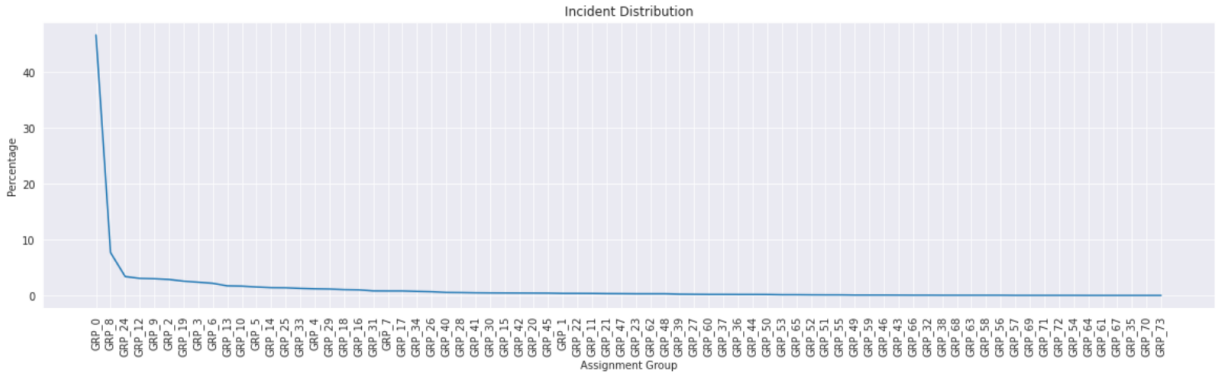

Figure 4: Ticket Assignment Visualization

It is a very imbalanced classification. In the end most of the groups do not even have more than 10 tickets. Hence it is decided to remove them to avoid the model being affected by them. After this, the dataset becomes (8321, 4). Instead of 74, there are only 49 groups left.

6. Feature Engineering

As we can see that both “Short description” and “Description” columns are text input by users. Hence they both should be considered to train the NLP algorithm. What I did here is simply to combine both columns as a new column “RawDescription” and only use the same for pre-processing and model training.

7. Handling of Special Chars & Numbers

To start with, the whole corpus of “RawDescription” is handled in a way that only characters (upper and lower cases) are kept. Special characters and numbers are replaced with space. This is because Special characters and numbers do not have meanings in this case and will not impact the classification. So we discarded them to avoid the model being influenced.

8. Lemmatization

Stemming would cause invalid words, hence we decided to build a lemmatization function and apply to “RawDescription” column. An example of the outcome looks like this:

Before: “login issue verified user details employee manager name checked the user name in ad and reset the password advised the user to login and check caller confirmed that he was able to login issue resolved”

After: “login issue verified user detail employee manager name check the user name in ad and reset the password advise the user to login and check caller confirm that be able to login issue resolve”

- Details -> detail
- Checked -> check
- Advised -> advise
- Confirmed -> confirm
- Was -> be
- Resolved -> resolve

9. WordCloud

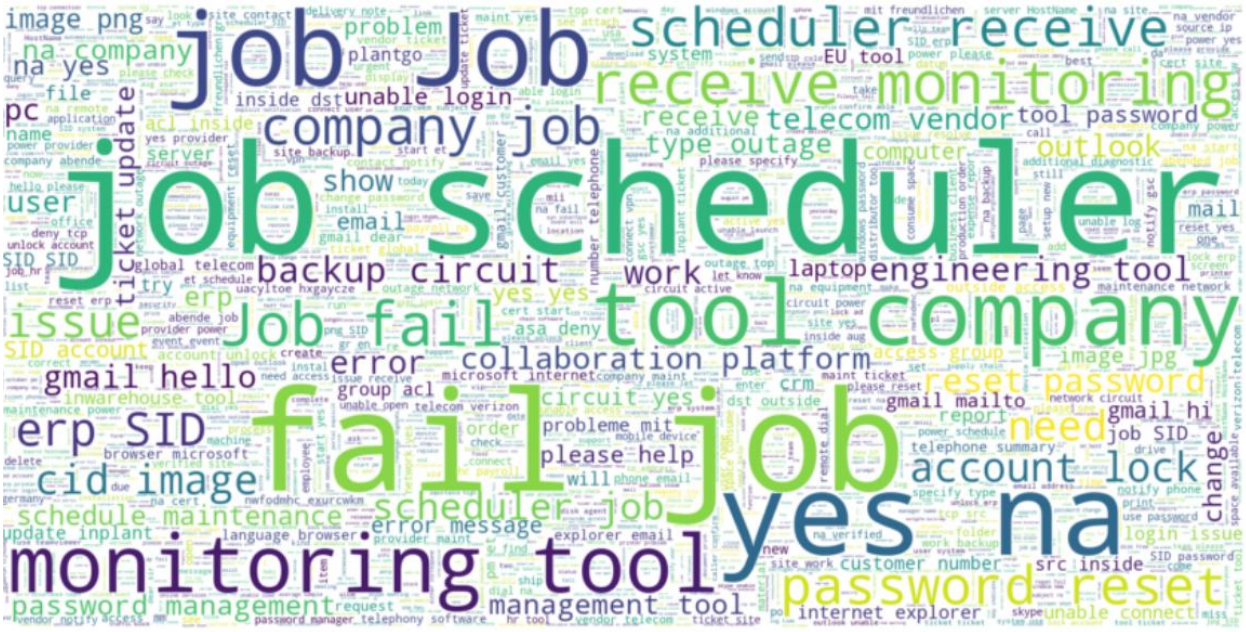

Figure 5: WordCloud

As we can see, If we look at the ticket descriptions overall, most of the tickets are for job scheduler fail, password reset, monitoring tool, password reset, etc..

10. Translate Assignment Group into Numerical Groups

We need to make groups numerical values to build the model. “Categorical” and “Factorize” are used in this case.

11. TF-IDF

TF-IDF is used to try out a few other classification models to start with. Here we select KNN and SVM, the performance of which will be compared with LSTM. The results are tuned.

Model	Training Accuracy	Testing Accuracy
KNN	67.95%	64.87%
SVM	88.81%	68.04%
Bagging Classifier	73.86%	58.15%

There is severe overfitting for SVM and Bagging Classifier; for KNN the model fits well with an accuracy of ~60%. “model_selection.cross_val_score” is used to evaluate the performance of the models by cross-validation. We get the mean and std of the results as below:

Model	Results_mean	Results_std
KNN	0.34	0.03
SVM	0.68	0.01
Bagging Classifier	0.63	0.02

To visualize it using boxplot:

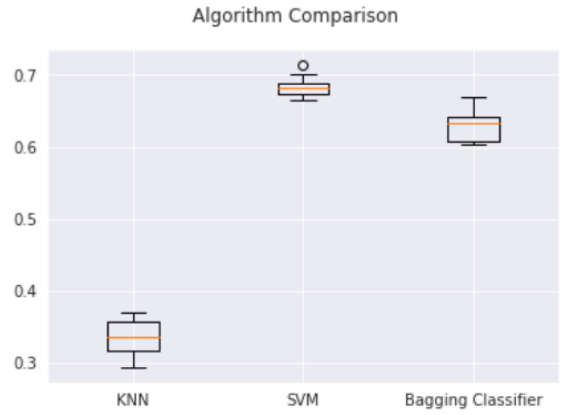

Figure 6: Boxplot of Classifiers

As we can see, SVM has the best performance. In our past practices, SVM does perform better than other classification models in many cases.

The reason why these algorithms is included in the project is that it is wondered in which conditions, or under which constraints, deep learning should be used instead of simpler models. Observation is that if your data available is sequential (every sentence is in a form of sequence of words), then LSTMs will do the job because it works with word embeddings but the training will take much longer. It is good at remembering the past sequences in the text and coupling it

with pretrained Wrod2Vec; but if it is a basic text classification then SMV or other supervised learning algorithms would be enough.

12. LSTM

Before building a model, tokenizer and text_to_sequences is used to index the texts. A new column is created to see the result:

	Short description	Description	Caller	Assignment group	len	RawDescription	group	token_text_vocab
0	login issue	-verified user details.(employee# & manager name)\r\n-checked the user name in ad and reset the password.\r\n-advised the user to login and check.\r\n-caller confirmed that he was able to login.\r\n-issue resolved.	spxjnwir pjlcoqds	GRP_0	29	login issue verified user detail employee manager name check the user name in ad and reset the password advise the user to login and check caller confirm that be able to login issue resolve	0	[53, 32, 303, 30, 176, 206, 135, 59, 105, 3, 30, 59, 4, 243, 14, 31, 3, 18, 281, 3, 30, 1, 53, 14, 105, 410, 258, 55, 2, 96, 1, 53, 32, 187]
1	outlook	\r\n\r\nreceived from: hmjdrvpb.komuaywn@gmail.com\r\n\r\nhello team,\r\n\r\nmy meetings/skype meetings etc are not appearing in my outlook calendar, can somebody please advise how to correct this? \r\n\r\nkind	hmjdrvpb komuaywn	GRP_0	23	outlook receive from hmjdrvpb komuaywn gmail com hello team meeting skype meeting etc be not appear in outlook calendar can somebody please advise how to correct this kind	0	[50, 17, 7, 3402, 3403, 20, 6, 70, 112, 346, 94, 346, 456, 2, 8, 440, 4, 50, 663, 36, 5258, 21, 281, 324, 1, 385, 33, 509]
2	cant log in to vpn	\r\n\r\nreceived from: eylqgodm.ybqkwiam@gmail.com\r\n\r\nhi\r\n\r\ni cannot log on to vpn\r\n\r\nbest	eylqgodm ybqkwiam	GRP_0	9	can not log in to vpn receive from eylqgodm ybqkwiam gmail com hi i can not log on to vpn good	0	[36, 8, 85, 4, 1, 92, 17, 7, 4200, 4201, 20, 6, 128, 15, 36, 8, 85, 11, 1, 92, 195]

Figure 7: Indexed Words

In “token_text_vocab” column, we can see that all words are indexed. The lower the index number is, the more frequent the word has appeared in the corpus.

Vocabulary size is 14105. Top 10 words are:

Index	Word
1	To
2	Be
3	The
4	In
5	Job
6	Com
7	From
8	Not
9	No
10	a

“./glove.6B.200d.txt” is imported as the embedding and saved in the matrix.

Training and Testing dataset are divided with a 0.2 ratio. Their sizes are:

Data set	Size
X_train	(6656, 1572)
X_test	(1665, 1572)
y_train	(6656, 49)
y_test	(1665, 49)

1572 is the max length of the text. 49 is the number of groups we kept.

This means that we are training on 6656 entities and test on 1665 entities. The input is the padded sequence of all texts (for those shorter than 1572, they will be post padded with 0). The output is the probability that each entity is falling into all 49 groups.

Some parameters: epochs = 10; batch_size = 64; embedding_size = 200.

A bi-directional LSTM model is built as below:

```
model = Sequential()
model.add(Embedding(input_dim=num_words,
                    output_dim=embedding_size,
                    weights=[embedding_matrix],
                    input_length=maxlen,
#                   mask_zero=True,
                    trainable=False))
model.add(SpatialDropout1D(0.2))
model.add(Bidirectional(LSTM(100, dropout=0.2, recurrent_dropout=0.2)))
model.add(Dense(100, activation='relu'))
model.add(Dropout(0.1))
model.add(Dense(49, activation='softmax'))
```

Figure 8: Bi-Directional

Embedding layer followed by LSTM units is applied. Softmax comes in the last for classification.

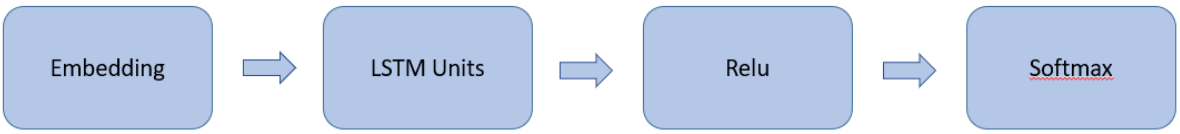

Figure 9: LSTM Strusture

Results of the model:

Epoch	Accuracy	Validation accuracy
1	0.56	0.57
2	0.58	0.59
3	0.60	0.60
4	0.62	0.61
5	0.64	0.62
6	0.65	0.64
7	0.66	0.64
8	0.67	0.65

9	0.68	0.65
10	0.70	0.65

13. Tuning of Hyperparameters

13.1.Tuning of Batch Size

Batch size is a hyperparameter that defines the number of samples to work through before updating the internal model parameters. It controls how often to update the weights of the network. It must be a factor of the training / testing size. Now we keep epoch constant and tune batch size to 128 but the accuracy apparently is lower than above (each epoch the accuracy is ~0.2 less). Then we tune to batch size = 64, the results are almost the same.

Results of the model:

Epoch	Accuracy	Validation accuracy
1	0.51	0.56
2	0.58	0.58
3	0.60	0.61
4	0.62	0.61
5	0.64	0.61
6	0.65	0.63
7	0.66	0.63
8	0.67	0.63
9	0.68	0.64
10	0.69	0.65

We will keep batch size = 64 because it has the best result while saves training time.

13.2.Tuning of Epoch

a hyperparameter that defines the number times that the learning algorithm will work through the entire training dataset. To improve the accuracy, we can try to increase the epoch number. We keep the batch size = 64 and tune Epoch to 100. The accuracy did not improve and takes very long to train.

After hyperparameter tuning, we decided to keep batch size as 64, and epoch size as 10.

14. Understand the Performance

Now that we have decided a relatively good parameter combination, a visualization is plotted together with a classification report to understand the performance of the model.

Final Result after prediction:

Model	Training Accuracy	Testing Accuracy
Bi-LSTM	74.22%	65.11%

Model Accuracy:

Figure 10: Model Accuracy

Model Loss:

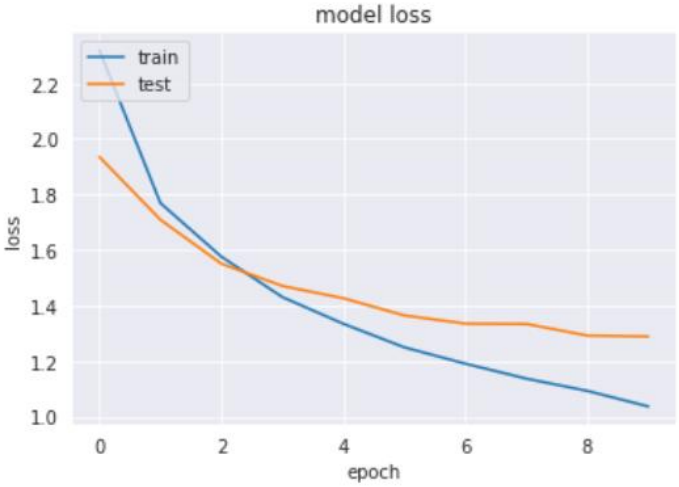

Figure 11: Model Loss

Confusion matrix:

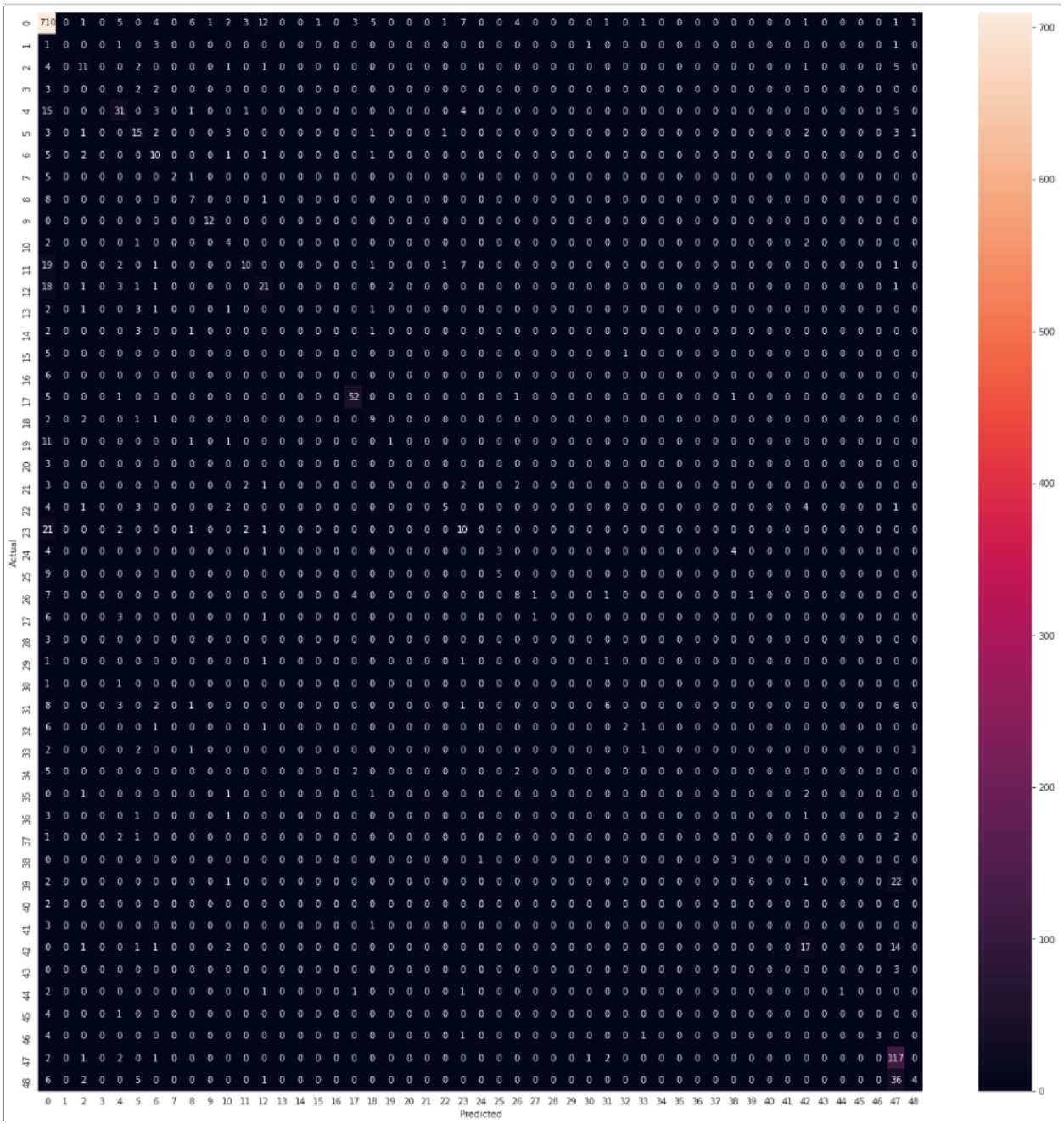

Figure 12: Confusion Matrix

The diagonal elements represent the number of points for which the predicted label is equal to the true label, while off-diagonal elements are those that are mislabeled by the classifier.

The higher the diagonal values of the confusion matrix the better, indicating many correct predictions. As we can see that many Assignment groups are not present in the test data. The diagonal element value for GRP_0 is highest.

Classification report is as below:

Classification report:

	precision	recall	f1-score	support
0	0.76	0.92	0.83	770
1	0.00	0.00	0.00	7
2	0.44	0.44	0.44	25
3	0.00	0.00	0.00	7
4	0.54	0.52	0.53	60
5	0.37	0.47	0.41	32
6	0.30	0.50	0.38	20
7	1.00	0.25	0.40	8
8	0.35	0.44	0.39	16
9	0.92	1.00	0.96	12
10	0.20	0.44	0.28	9
11	0.56	0.24	0.33	42
12	0.48	0.44	0.46	48
13	0.00	0.00	0.00	9
14	0.00	0.00	0.00	7
15	0.00	0.00	0.00	6
16	0.00	0.00	0.00	6
17	0.84	0.87	0.85	60
18	0.43	0.60	0.50	15
19	0.33	0.07	0.12	14
20	0.00	0.00	0.00	3
21	0.00	0.00	0.00	10
22	0.62	0.25	0.36	20
23	0.29	0.27	0.28	37
24	0.00	0.00	0.00	12
25	0.62	0.36	0.45	14
26	0.47	0.36	0.41	22
27	0.50	0.09	0.15	11
28	0.00	0.00	0.00	3
29	0.00	0.00	0.00	4
30	0.00	0.00	0.00	2
31	0.55	0.22	0.32	27
32	0.67	0.18	0.29	11
33	0.25	0.14	0.18	7
34	0.00	0.00	0.00	9
35	0.00	0.00	0.00	5
36	0.00	0.00	0.00	8
37	0.00	0.00	0.00	6
38	0.38	0.75	0.50	4
39	0.86	0.19	0.31	32
40	0.00	0.00	0.00	2
41	0.00	0.00	0.00	4
42	0.55	0.47	0.51	36
43	0.00	0.00	0.00	3
44	1.00	0.17	0.29	6
45	0.00	0.00	0.00	5
46	1.00	0.33	0.50	9
47	0.53	0.93	0.68	126
48	0.57	0.07	0.13	54
accuracy			0.65	1665
macro avg	0.33	0.24	0.25	1665
weighted avg	0.61	0.65	0.60	1665

Figure 13: Classification Report

For uneven classification distribution, f1-score is more representative.

F1 Score = 2*(Recall * Precision) / (Recall + Precision).

In this case, we can see that f1-score for GRP_0 is the highest. For some groups that contain very few entities, the score is very low.

Model summary:

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 1572, 200)         2803000
_________________________________________________________________
spatial_dropout1d_1 (Spatial (None, 1572, 200)         0
_________________________________________________________________
bidirectional_1 (Bidirection (None, 200)               240800
_________________________________________________________________
dense_2 (Dense)              (None, 100)               20100
_________________________________________________________________
dropout_1 (Dropout)          (None, 100)               0
_________________________________________________________________
dense_3 (Dense)              (None, 49)                4949
=================================================================
Total params: 3,068,849
Trainable params: 265,849
Non-trainable params: 2,803,000
_________________________________________________________________
None
```

Figure 14: Model Summary

To see how it reflects, we test the model on a new ticket.

Ticket = ''I have issue fixing the network problem which was working fine last month'.

Passing through the model, we got below result:

```
Ticket : [[15 26 32 ...  0  0  0]]
Output: [[3.48058581e-01 2.59029912e-04 7.99549103e-04 1.25974300e-03
  2.06175935e-03 1.72493642e-03 1.31917803e-03 2.66252144e-04
  1.14151845e-02 1.31802226e-04 7.20456301e-04 1.19521491e-01
  2.79806624e-03 4.73260821e-04 7.27224164e-03 5.08432975e-04
  7.18264747e-03 3.88154178e-04 4.33466816e-03 5.88646578e-03
  7.11645093e-03 7.42796343e-03 1.13814324e-03 3.66924614e-01
  1.07744127e-03 1.02933291e-02 1.20641117e-03 1.09987555e-03
  1.98174897e-03 6.95853261e-03 1.69281324e-04 2.24693920e-02
  9.92655521e-04 5.45088062e-03 4.32484492e-04 7.53762608e-04
  1.96797680e-03 1.53057976e-04 7.44577206e-04 8.81582790e-04
  4.14699921e-03 4.63109463e-04 2.30036472e-04 7.08509528e-04
  2.71967240e-03 1.76724454e-03 2.66002622e-02 1.86404644e-03
  5.87805919e-03]]
```

Figure 15: Test Result for New Ticket

Ticket is the input and the output shows the possibility that falls under each group. We can tell that it is most likely to fall under GRP_24 with 3.66e-01 probability.

Insight, Recommendation & Lessons Learned

1. The performance of a machine learning model not only depends on the model, but greatly depends on the dataset – the cleanness, completeness and proper handling of consistencies. This is discovered because in Milestone 2, a similar model was used but the accuracy is merely 45% because the author missed out on a few steps in data handling. It is important to do thorough EDA before any model is implemented

2. LSTM usually outperforms most of the other models when we want our model to learn from long term dependencies (sequential texts like in this use case). Final comparison is as below. SVM as well does a good job but with overfitting. The author still recommends Bi-LSTM model for the business case, because as the input increases and the diversity increases, simple models like SVM may not maintain such performance.

Model	Training Accuracy	Testing Accuracy
KNN	67.95%	64.87%
SVM	88.81%	68.04%
Bagging Classifier	73.86%	58.15%
Bi-LSTM (after tuning)	74.22%	65.11%

3. Hyperparameter tuning requires the approach to be structured and consistent. For example, tuning of batch size should keep epoch and other parameters unchanged. This is discovered because in Milestone 2, many times of tuning led to no change due to the fact that they were changed at the same time.

4. This is a highly applicable algorithm. To relate with the problem statement, it resolves:
    a. It is time-consuming to manually perform ticket allocation – **with the application of this algorithm, we no longer need human labor to manually read and classify the tickets. With more samples and historical data available, the accuracy of the model will improve greatly and the model will be more accurate and mature.**
    b. It is costly to hire more workforces to perform this allocation and train them – **the algorithm will require initial investment to purchase and implement. However, once it is widely used, cost of employment in such function will decrease. It is advised that corporates implement it for certain business units first, start small, start stable.**
    c. There will be unavoidable mistakes in this allocation due to operational risk / lack of knowledge / subjective understanding, etc. – **One of the most important drivers of end-to-end automation is to guarantee there is no bias or operational risk. After this algorithm is implemented, it could be as well integrated with other functionalities like RPA in data transportation, approval, etc..**
    d. Bad employee user experience due to above three concerns – **Employees will be re-purposed to other more business-related functions and better user experience is guaranteed. Users who raised tickets will received a more responsive feedback with higher effectiveness.**

5. Areas of improvement
    a. Short description is a summary of description. In this use case, we used feature engineering because both texts are written by human. In the future, short description could be implemented as pre-defined dropdown list where users are able to search and choose. Certain values will route to certain L1 support on a rule-based orchestration, and the algorithm could be applied later. This saves infrastructure and implementation investment, and in future enhancement, it is easier to integrate with RPA.
    b. Language differences shall be minimized. Similar to above point, it should be routed to certain country-specific portals to avoid too much different data to be mixed together.
    c. Implement a feedback session with users who raised the ticket. For better implementation, a feedback program is important to learn how effectively the model is behaving. Users could feedback whether they are satisfied with the experience and whether they have spoken to the expert required – this could incorporate in the supervised learning and further improve the model performance.
    d. Even though the performance is acceptable, but to better implement in real world, more data is encouraged to be collected. For example, "Urgency" could vary from 1-5 so we can count it in the algorithm. When we add more data, the more structured, the better. Meanwhile, 500 more records for each group is encouraged to be collected.

Appendix

Capstone_Final_Submission.ipynb

Appendix 1: Source Code