Automating Bill of Lading Creation

IE3100R/IE3100M Systems Design Project (AY2019/2020)
Department of Industrial Systems Engineering and Management
Industrial Supervisor: Mr. Kevin Deans
Academic Supervisors: Prof. Andrew Lim, Prof. Cao Zhiguang
SDP Group 8 Members: Jeremy Lee Jun Wei, Kan Xue Yu, Le Khang Tai, Sim Khiang Leon

PROJECT ACHIEVEMENTS

- Fully trained model to extract required entities in existing client pool
- Multiple layers of fail-safe protocol to ensure data integrity, correctness and system resilience
- End-to-end executable script to extract and submit data into an online server (process automation)

SKILL SETS

- Machine Learning (Natural Language Processing / Named Entity Recognition)
- Python Programming
- Optical Character Recognition (OCR)
- Deep Learning / Neural Networks

- Systems Thinking
- Process Flow Diagrams
- Project Management
- Swiss Cheese model of system accidents

- Multiple Criteria Decision Analysis (MCDA)
- Modelling and Analytics
- Human Factors Engineering
- Robotic Process Automation (RPA)

Company Background

Leo Shipping Agencies is a service-oriented company that focuses on providing agent services to clients whose vessels call at Singapore. The mission of the company is to provide a comprehensive range of port and liner agent services for ship owners worldwide, through a responsive and knowledgeable team, up-to-date technology and a network of shipping fraternity in the maritime industry. The vision of the company is to strive to provide quality and competitive tailor-made solutions to every customer according to their logistical requirements. Core values of the company consists of dedication and commitment, as well as being focused and knowledgeable.

Problem Description

- Automation of the generation of the Bill of Lading (B/L)
- B/L: legal document provided by a carrier to a shipper that serves as a shipment receipt when the carrier delivers the goods at a predetermined destination
- Name entity recognition (NER) problem (~15 entities to recognize and extract)
- Large variety of different B/L layouts in different formats and file types provided as data

Objective

The project aims to automate the process of extracting required information from the B/L drafts and inserting it into an online server, while satisfying the following objectives:

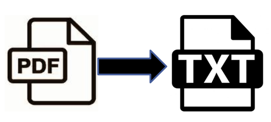
- Company Objectives (reduction in processing time, human interventions and complexity of process)
- Academic Objectives (effectively applying ISE technical skill sets in problem solving)
- Other Objectives (accuracy of information retrieval, scalability of product delivered)

Methodology

Preprocessing Data

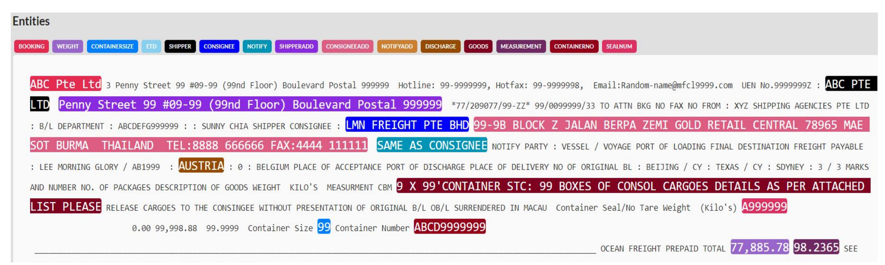
Step 1: Converting PDF files to text files

Text are extracted from native PDFs using a Python library called *pdf2image*. Scanned PDF documents have to be converted to images before being processed by tesseract OCR to extract the text. The quality of the text that is extracted from the OCR depends on the condition of the scanned document. The output resemble a string of unorganised words in a paragraph and the original PDF layout is not retained.



Step 2: Annotate entities of the text files

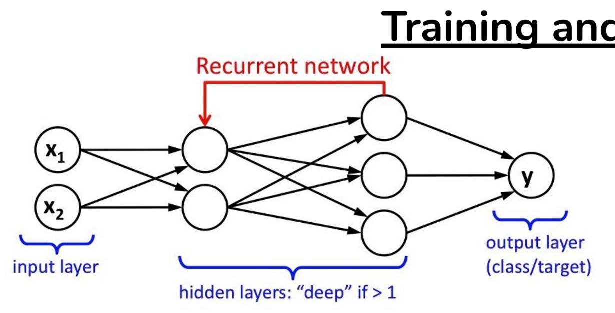
There are more than 15 entities that are required to be extracted from a B/L draft. This has to be done for all the training data. The file will be saved in JSON format.



Training and Testing of Model

Step 3: Build and Train a Recurrent Neural Network Model

Spacy library is being used here for building the Neural Network model for text recognition. It is a free open-source library commonly used to solve Natural Language Processing problems. Optimal hyperparameters chosen for the problem: Iteration = 500, decaying dropout = 0.5, optimizer = SGD.



Step 4: Test and Validate the Model

Performance of models are judge based on NER training loss along with precision and recall percentages of testing data. The lower the NER loss, the better the model is at extracting the correct information. The higher the precision and recall, the more desirable the model performance. Repeated testing, improving and retraining models are done in search for the best model.

Extracting and testing entities

Step 5: Extracting entities using best model

Entities extracted by the model will be put into an excel file. This excel template was formulated after consulting and validating with veteran staffs in the company. It contains all the essential information for the generation of B/L.

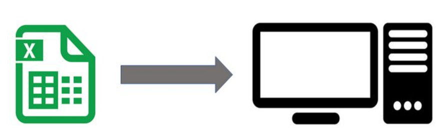
Step 6: Fail-safes of extracted entities

After entities are extracted it will run through multiple layers of checks to ensure that the extracted fields are not empty and extracted information are of the correct data type. If any of the checks fail, error prompts will be activated to request human intervention. Certain entities corrected by human intervention will be stored in a separate file to act as a database to be used for improving future fail-safes.

Inputting data into server

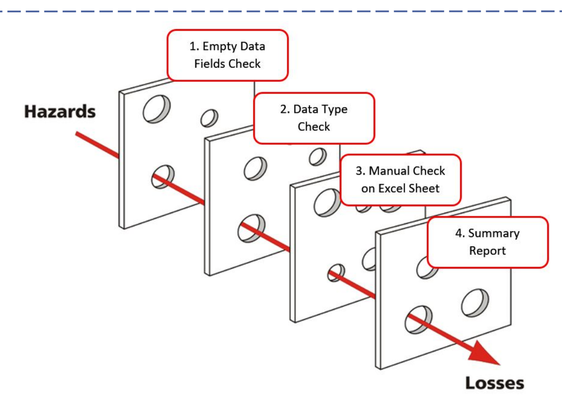
Step 7: Robotic Process Automation to transfer extracted information into online server

This was created using *Pyautogui*. The script mimics the keyboard and mouse movements to replicate the actions that the user would do when filling up details to the front-end of the server. Information will be transferred from the completed excel sheet from step 6 to the server. After every successful transfer of data per B/L, information is archived for the respective B/L for future reference.

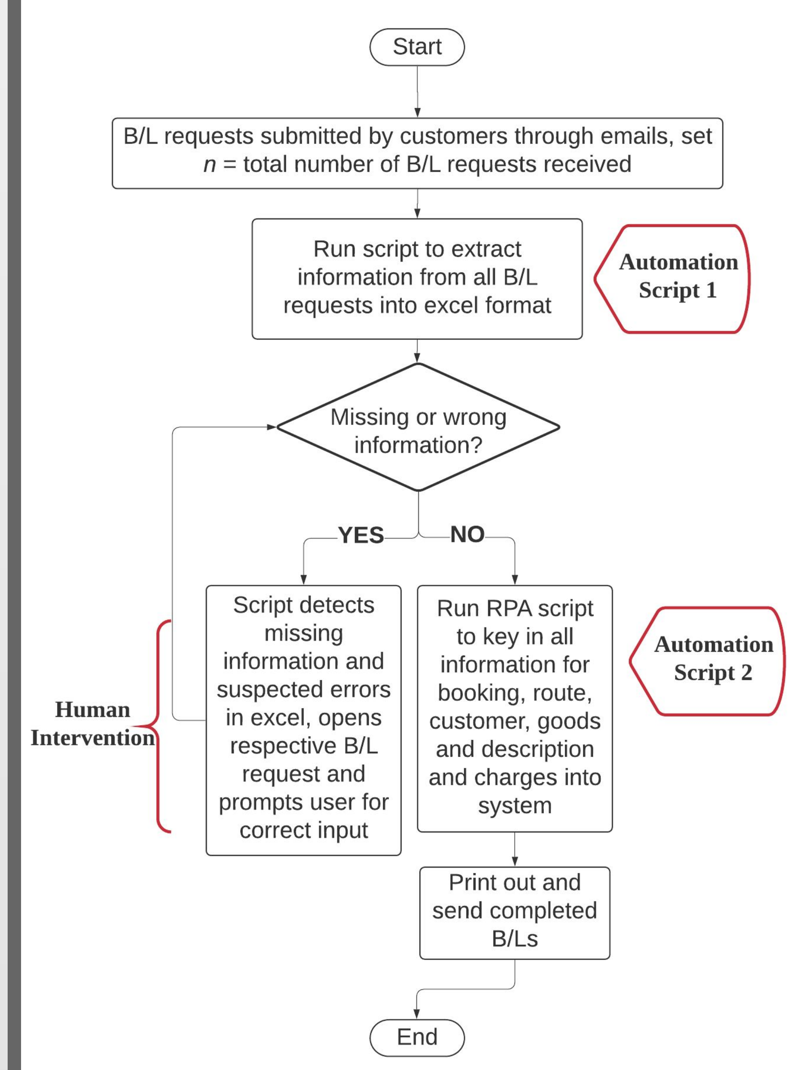


Step 8: Report summary status of input process

At the end of the script in step 7, a summary report will be saved as a .txt file to reflect any potential errors that has been committed. This .txt file will be shown to the staff to remind him/her of the details that could be filled in erroneously. This would reduce the time needed for the staff to backtrace should there be any inaccuracies in data submitted.



Process Flow



Comparison and Critique of Design Alternatives

Brute-Force Approach

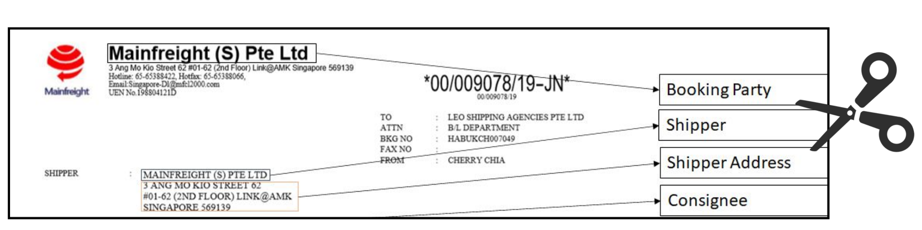
Description

Information is extracted by hardcoding fixed coordinates and location of every entity found in a B/L draft, with the assumption that the layout does not change for each individual client's B/L draft. The approach could be tailored for every client, producing a script dedicated to each client to be run when the client submits a B/L request.

Evaluation

- Tedious process to do hardcoding for every client's B/L draft
- Does not cater for new companies
- Not robust to deal with human error and even slight layout changes

Approach is attempted but not reliable as a standalone solution. As such, the solution will not be fully adopted and will be used instead to aid the machine learning approach.



Pure Machine Learning Approach

Description

A Natural Language Processing model from Spacy will be adopted and re-trained with suitable datasets provided by the company. The objective of this approach is to evaluate and exploit hidden dependence that may exist between different clients' B/L drafts, and from there, pick the best performing model.

Experimental Set-up

1. Annotate existing client's B/L draft and create replication of the dataset with key value being changed
2. Split training and testing dataset with an 8:2 ratio
3. Set up the model with the following hyperparameters: Iteration = 500, dropout = 0.5, optimizer = SGD
4. Train the model with a single client's data produced in step 1
5. Test the model's performance using performance metrics: precision and recall
6. Save the model separately
7. Repeat step 1-5 for other clients
8. Set up a new model with the same hyperparameters and train it with the combined dataset of all clients
9. Test the model from step 7 using the same performance metrics
10. Compare and evaluate the results

$$Precision = \frac{True\ Positive}{Actual\ Results}$$

$$Recall = \frac{True\ Positive}{Predict\ Results}$$

Results

	Tested with learnt B/L layout				Tested with new B/L layout	
Model	Overall Precision		Overall Recall		Overall Precision	Overall Recall
	Training	Testing	Training	Testing		
MainFreight	99.37%	92.31%	99.38%	92.31%	30.77%	30.77%
Shipco	84.79%	83.33%	84.78%	84.68%	40.44%	40.37%
Hock Cheong	98.44%	92.31%	98.31%	81.25%	39.68%	40.47%
Combined	99.37%	92.31%	99.38%	92.31%	66.17%	65.28%
	84.79%	91.66%	84.78%	92.33%		
	98.44%	92.86%	98.31%	87.25%		

Evaluation

The combined model performs either as good or better than individual models. Drastic reduction in performance was observed when the model is tested on a new dataset even if the new dataset is visually similar to an the existing dataset. Relying solely on a machine learning model to extract data results in severe inaccuracies. This approach is therefore not reliable. A better solution would be to build on this approach and attempt to cover more grounds by training on more variety of B/L drafts. A better solution should also be able to find ways to deal with new B/L drafts despite not able to extract any information.

Results

Performance of Chosen Model (Precision & Recall)

Discussion & Evaluation

Firstly, the initial performance of the trained model will be evaluated. Secondly, the performance of the model that uses both Machine Learning and regular expression hardcoding will be evaluated. Lastly, the performance of the model that has conducted the fail-safe checks with human intervention will be evaluated.

In Table 1: The performance of individual companies range from 40%~100% which resulted in the overall performance of approximately 85%. From the test, it is observed that certain companies exhibit the same type of error which allows room for manual extraction to solve such error types.

In Table 2: Manual extraction efforts were focused on top clients which are listed in table 2. Manual extraction using regular expression hardcoding has increased the performance of the model, even for Tricon's B/L drafts which are usually seen in a scanned format and has consistently poor performance from trained model.

In Table 3: The values reflects the overall performance of the model after manual extraction and fail-safes. It can be seen that the overall performance had increase by approximately 10%. The reason why the performance will not reach 100% is because of the fact that there are still some errors that could not be captured by the script. An example of that would be misspelled names or awkward spacing.

Overall, the performance have drastically improve after manual extraction and fail-safes. This further proves the effectiveness of the automation approach.

Model	Precision	Recall
Overall	84.31%	86.67%

Table 1: Trained Model Standalone Performance

Clients	Overall Recall	
Modification	Before	After
Shipco	84.78%	97.86%
Hock Cheong	96.44%	98.31%
Tricon	63.46%	93.24%

Table 2: Performance after manual extraction

Model	Precision	Recall
Overall	93.31%	94.55%

Table 3: Performance after Fail-Safes

New Approach vs Original Approach

Performance Metrics

- Number of interventions
- Time spent by user

Discussion & Evaluation

To further measure efficiency of the chosen approach, three types of time analysis were conducted. Automation allows the staff to continue working on other tasks while the executable scripts run. As a result, only the time spent during intervention will be considered in the calculations and not the entire run time of the scripts. The control experiment will be calculating the average time spent on manual labour of the original process without automation. This is the average time that the staff spent creating a B/L from a B/L draft manually. This was recorded to be 3min 54s = 234s.

1st analysis (Time taken for best-case scenario): For the best case scenario, 3 B/L drafts from each of the top 3 clients were used for this analysis as they require the least number of prompts. Results are in Table 4. Percentage of time saved was 78.06%.

2nd analysis (Time taken on average): Average number of prompts for each B/L draft was found to be 7. Analysis was then done on 5 randomly chosen B/L drafts that requires 7 human interventions. Results are in Table 5. Percentage of time saved was 65.28%.

3rd analysis (Time taken for worst-case scenario): For this analysis, 5 B/L drafts were used for which the user has to key in all 15 data fields, through 15 error prompts per B/L. Results are in Table 6. Percentage of time saved is 26.48%.

Clients	Hock Cheong			Shipco			Tricon			Avg
B/L	1	2	3	1	2	3	1	2	3	
Prompts	5	4	4	3	3	3	6	5	5	4.2
Time (s)	60.12	44.51	49.31	36.39	34.67	32.48	75.11	68.17	61.02	51.31

Table 4: Best-case scenario

B/L	1	2	3	4	5	Avg
Time (s)	79.39	84.88	80.93	83.59	77.40	81.24

Table 5: Average scenario

B/L	1	2	3	4	5	Avg
Time (s)	175.65	177.80	174.89	168.77	163.03	172.03

Table 6: Worst-case scenario

Evaluation of Chosen Approach

The chosen approach is a hybrid of making use of brute force techniques along with a well-trained machine learning model. Separating the automation process into 2 parts, one executable script for information retrieval and one executable script for RPA, is a technique adopted to implement the additional layers of fail-safes in between to better capture potential errors and allow for user's easy access to edit the data extracted. This layered security feature also ensures that the chosen approach does not fail when encountering a B/L request with a new layout. Information extraction will still be attempted despite varied accuracy, and the respective errors will be captured and prompted. Human factors are also taken into consideration when prompting for human intervention. Error prompts and window popups to show the relevant document makes it convenient for user reference when correcting data. Usability testing is done with the company's staff and the feedback has also been integrated to further improve the system. All in all, the various objectives have been met and a large degree of automation has been achieved.

Recommendations

It is recommended that the company adopt the proposed solution to replace their existing protocol as it has been proven to be more time effective if the script is being used. Even in the scenario where no entities were extracted, the performance in terms of time saved have improved by approximately 20%.

In the future, the project can further improved or expanded in the follow directions:

- Modified for other similar natured projects (Retrain model using different datasets)
- Project could be improved to cater to other formats of data
- As more information gets extracted from new B/L drafts in the future, this information can serve as new training datasets for the current model. The current approach can be further automated to retrain the model by itself to further enhance the performance of the model.
- New features could be introduced to increase usability of the system