Dmitry Khramov

Robotic and machine learning: How to help support to process customer tickets more effectively

Metropolia University of Applied Sciences

Bachelor of Engineering

Information Technology

Thesis

12 April 2018

Metropolia

Author Title Number of Pages Date	Dmitry Khramov Robotic and machine learning: How to help support to process customer tickets more effectively 41 pages 12 April 2018
Degree	Bachelor of Engineering
Degree Programme	Information technology
Professional Major	Software engineering
Instructors	Marko Lähde, Senior Solution Manager, Basware Oy Peter Hjort, Senior Lecturer

The objective of the thesis work was to explore opportunities provided by emerging digital technologies, such as robotic process automation and machine learning and apply these technologies for enhancing the efficiency of company's business process. Developed automation was embedded into existing process of handling incoming customer issue tickets. It allowed a significant increase in the tickets' processing speed and will relieve human workforce from repetitive, low value-added tasks.

Process automation was combined with machine learning in order to introduce decision making capabilities for the robotics workflow. Historical data from issue tickets was used to train supervised machine learning algorithm for prediction, based on textual content. This method indicated whether a ticket belongs to the particular group. Several classification algorithms were evaluated by accuracy performance with different representations of textual data. Superior performance results were achieved by using linear support vector machines and logistic regression which were trained on the data. Finally, the results were transformed into binary vector representation with n-gram range from 1 to 3.

Implemented automation identifies correct support group with 75% accuracy and 76% precision, which allowed taking under RPA coverage 16% of English subset of processed tickets. At the time of writing this study, the automation was evaluated in test environment and considered for further extensions and accuracy improvements.

RPA powered by machine learning enables to overcome the limitations of plain rule-based robotics automation. Also, it provides potential for developing sophisticated, intelligent automations, capable of handling a higher range of business processes.

Keywords	robotics process automation, RPA, machine learning, text classification

Contents

List of Abbreviations

AI	Artificial Intelligence.
API	Application Programming Interface.
AWS	Amazon Web Service.
BPM	Business Process Management.
CRM	Customer Relationship Management.
FTE	Full-time Equivalent.
IDF	Inverse Document Frequency.
ITSM	Issue Tracking System Management.
LSI	Latent Semantic Indexing.
PCA	Principal Component Analysis.
RPA	Robotic Process Automation.
REST	Representational State Transfer.
SVM	Support Vector Machines.
TF	Term Frequency.
TF/IDF	Term Frequency - Inverse Document Frequency.
XML	Extensible Markup Language

1 Introduction

The modern trend of software automation in business is driving a fundamental change in the approach companies and employees interact with their customers and each other. Computing power and data access have released the opportunity to consider a new reality where traditional organizational structures are complemented by a robotic workforce.

Intelligent automation systems can detect and produce extensive amount of information and automate entire workflows or processes, learning and adapting themselves. Applications are varied from data collection, analysis and decision making to guiding autonomous vehicles and advanced robots.

Until recently, most of the robotics applications were used in the primary sector, automating and replacing the human component from the production chain. Today, tertiary business sectors have already started to apply new technologies and the robotic paradigm in order to automate their processes and remove humans from low value-added activities. This form of digital transformation is already helping companies transcend conventional performance trade-offs to achieve remarkable levels of efficiency and quality, and showing a significant return on investment.

The research is implemented for the case company Basware Oy, which provides enterprise software for managing financial processes. The objective of the topic is to explore and analyze how robotics and machine learning could be used to improve the customer ticket processing and implement the proposed solution based on theory and business requirements.

The thesis contains 6 sections. Following the Introduction, Section 2 presents the relevant theory behind the topic. Section 3 contains the analysis of the actual state in the case company, regarding the selected subject. Section 4 explains which technologies and methods were used to achieve the objective. Section 5 focuses on implementation of the solution, while Section 6 summarizes thesis results and presents recommendations for further development.

2 Theoretical background

Current section focuses on the theory behind RPA and machine learning concepts. First, it provides the overview of RPA, its features, use cases and benefits. Then, the process of RPA implementation and its challenges are described. Machine learning techniques, data preprocessing methods and text classification approaches are explained in the latter subsections.

2.1 RPA background

According to the Institute for Robotic Process Automation (IRPA) RPA is the application of technology that allows employees in a company to configure computer software or a "robot" to capture and interpret existing applications for processing a transaction, manipulating data, triggering responses, and communicating with other digital systems (Institute for Robotic Process Automation, 2015). In other words, RPA is the technological imitation of a human worker whose goal is to deal with structured tasks in a fast and cost-efficient manner. It is implemented with a software robot, which mimics human behavior, while interacting with different software via front-end. (Asatiani and Esko, 2016)

Three main characteristics of RPA allow distinguishing it from alternative automation approaches such as Business Process Management (BPM), scripting and screen scraping. First of all, RPA can be configured effortlessly, without requirement for high level programming skills. User can implement complex workflows by simply dragging, dropping and linking icons that represents steps in a process. This distinguishes RPA from BPM, systematic approach for enhancing efficiency of organization's workflow, which requires knowledge of programing skills.

The second distinctive characteristic is that robotics technology performs on top of existing systems, without the need of developing or replacing expensive platforms. By operating with other systems through presentation layer, software robots do not communicate with system's Application Programming Interface (API) and do not require any change in underplaying programming logic. On the other hand, BPM solutions require creation of new applications, access to business logic and data layers in the IT architecture stack.

The last feature, describes RPA as a robust platform that is designed to meet enterprise IT requirements for security, scalability, auditability and change management. RPA robots are deployed, scheduled and monitored on centralized, interconnected IT supported infrastructure to ensure transactional integrity, compliance with enterprise security models and continuity of service in line with the enterprise' business continuity plans. In comparison, scripting and screen scraping methods, used to capture keystrokes and mouse clicks and deployed locally on the desktops, could not provide such reliability. Screen scrapes are able to understand specific positions of the fields on the screen what makes them unusable if the fields were moved to another place and screen scraper has not been reconfigured. (Lacity and Willcocks, 2016)

Attended vs unattended automation

Two different approaches for automating business processes can be defined: semi-automation and fully automation. Semi-automation, also called attended automation, relies on interaction with a human operator to complete end-to-end task efficiently. Attended automation solutions are located at an operator's workstation and are triggered by specific events, actions or commands an employee undertakes within a specific workflow. Such kind of automation is required to be agile and user-friendly in order to provide employee opportunity to navigate between different interfaces and screens. (Ostdick Nick, 2017) For example, customer service desks in any financial institution have service agents navigating through multiple systems to view customers' data while interacting with them. Software robots can be used to help with collection of information from multiple source systems to support employees servicing client calls.

The other type is fully or, so called, unattended automation, which relies on principle of absence of human intervention. Automation robots trigger actions by themselves and work is executed continuously, without human intervention. That allows automation software to carry out actions on a 24/7/365 basis. This type of automation is generally used in the scenarios where massive amounts of information are being gathered, sorted, analyzed and distributed among stakeholders in an organization. For example, unattended RPA solution can perfectly serve business transactions in a health insurance company, which has a large amount of claims processing, invoices, and other documentation tasks. Events and actions within a workflow can be engaged by the automation robots themselves, which in turn promotes a more streamlined documentation and data management process. (Venkatesha and Kasmera, 2017)

In addition, unattended automation has an advantage over attended one, which resides on specific workstations and limits the access to the operator, currently engaging in a certain workflow in a specific workstation. On the other side, back-office automation allows remote access via a number of interfaces or platforms. Administrators can monitor, analyze, deploy and schedule automation processes, reporting and auditing business transactions in real-time within a centralized platform. This means employees have a greater capacity for collaboration and communication within an automation platform, which can help break down functional and communication silos in a cross-organizational manner. (Ostdick Nick, 2017)

RPA use cases

In order to determine, whether the business process is suitable for RPA, the following characteristics should be identified:

- Frequently performed tasks, which may include high volume of sub-tasks.
- Usage of multiple systems with a stable environment.
- Low complexity without subjective judgmental or creative thinking.
- Processes which can be detached into more simple, straightforward, rule-based steps.
- Tasks are prone to human mistakes. (Asatiani and Esko, 2016)

According to the mentioned above characteristics, wide range of industries can find benefits from utilizing RPA: legal, healthcare, insurance, utility companies, finance and banking. The following cases are some of the opportunities in the financial services industry:

- Setting up customer data – manual activity where operator invoke uploaded customer's documents and enters customer data into customer relationship management (CRM) system. RPA can be used for identification and entering customer information into CRM.
- Validation of existing customer information. RPA can perform this activity by accessing databases, extracting data from documents, collecting information from social media, merging data from different sources and filling in the forms.

- Customer information gathering. RPA can be used to gather, input and process structured and unstructured data.
- Customer servicing. RPA can help financial institutions to improve customer experience, increase operational speed and accuracy, navigate through huge amount of data, identify patterns, improve learning and accelerate decision making. (Venkatesha and Kasmera, 2017)

These are just few examples of possible opportunities where RPA can be applied. The figure 1 illustrates the dashboard with possible use cases for RPA utilization and related benefits.

Opportunity for RPA	Related benefits				
Use case	Enhanced accuracy and quality	Improved speed of operations	Increased staff productivity	Refined audit trail with accurate information	Increased time for strategic tasks
Validating existing customer information	✔	✔	✔	✔	✔
Documentation gathering	✔	✔	✔	✔	✔
Customer information gathering	NA	✔	✔	NA	✔
Compiling customer information	NA	✔	✔	NA	✔
Customer screening	✔	✔	✔	✔	✔
Customer servicing	✔	✔	✔	✔	✔
Regulatory monitoring and data collection	✔	✔	✔	NA	✔
Risk assessments	✔	✔	✔	✔	✔
Account closure processing	NA	✔	✔	✔	✔

Figure 1. Benefits of RPA across different use cases. (Venkatesha and Kasmera, 2017)

The figure above shows that various numbers of benefits can be achieved with integration of RPA into business processes. In order to obtain a better understanding of the impact of RPA, its advantages should be described in more details.

Benefits of RPA

As it was already mentioned, process automation applies specific technologies to automate routine, standardized tasks in support of an enterprise's knowledge workers. By freeing human employees from these day-to-day tasks to apply themselves to core business objectives, automation offers a number of irresistible benefits to the workplace.

Promoters of RPA frequently present it as a replacement for outsourcing. Routine, non-core processes, such as invoice processing, bookkeeping of data entry require many FTEs (full-time equivalent) resources. Organizations often try to outsource these routines to low-cost destinations, such as India. Although outsourcing helps to reduce employee costs and concentrate on core operations, many challenges to outsourcing take place, such as hidden cost of management, communication problems, and complex service level agreements. RPA guarantees not only to reduce costs (robots can work 24/7 without being paid), but also eliminate the problems with management and miscommunication. A typical software robot can cost an equivalent of 0.1 to 0.19 of an in-house FTE, and 0.33 to 0.5 of an offshore FTE. (Prangnell and Wright, 2015; Slaby, 2012)

RPA requires fully tracked and documented steps of the business processes, which makes company more compliant with industry and audit regulations. In addition, robots produce data, during tasks execution, which can be further analyzed. This provides better decision making in the field where processes are automated. Organizations are able to identify gaps where processes can be more optimized to increase efficiency. (Institute for Robotic Process Automation, 2015)

As it was mentioned earlier, existing IT systems are not needed to be changed in order to integrate RPA, as it mimics human behavior. Robots, which operate within user interface, provide considerable advantage compared with automation achieved through back-end integration, which requires serious changes of the exacting systems. (The Hackett Group, 2017)

One more potential benefit of RPA is not only moving people to more productive job, but in the long run it can create new jobs in robot management, consulting and data analysis.

2.2 RPA implementation roadmap

In most cases, after introducing a successful RPA pilot, organizations have difficulties finding the right path for a successful, sustainable roll out of an automation program. Without having knowledge how to move forward, which resources to mobilize, what kind of benefits expect and what mistakes to avoid or best practices to implement, companies face some sort of “chasm”. In order to bypass such kind of chasm and have some light on the issues they are encountered with, organization should have a clear understanding of the steps involved into full roll-out of RPA and be aware of most common pitfalls during this process. (Moayed, 2017)

Looking specifically at the automation process, the following seven steps can be distinguished:

- Process identification and prioritization. This step includes applying a methodology, which allows assessing the process for its RPA compatibility.
- Detailed process assessment. The second step involves examination of a process in more detail to see if the potential and complexity assessed during the first step still hold and what percent of this process can be actually automated.
- Process redesign. Commonly, when time to automation comes, it can be discovered that the processes are not standardized, optimized, documented or followed as they though. During this phase it is recommended to take an opportunity to optimize the process before proceeding to automation.
- Defining detailed user stories and business requirements. This is a crucial phase, which consists of description of the process in its most detailed steps and understanding as much as possible all the potential exceptions in order to develop robust RPA workflows that will be directed to RPA developers.
- Development. Based on previous step, actual RPA workflows are programmed and the process is automated.
- User acceptance testing. During this phase, the automated process is tested to observe its behavior and to repair potential bugs and catch possible exceptions that might be missed during previous steps.

- Hypercare. The recommendation is to monitor the process for the period of two weeks by the person who has developed the automation in order to correct any remaining issues that may appear until a high level of reliability is achieved. (Moayed, 2017)

In addition to the mentioned RPA implementation procedure, organizations need to have a higher level plan of RPA roll-out which should include several key dimensions.

First of all, organizations, especially large international companies, should take into consideration the scope of the RPA roll-out. Companies should decide which functions should be included, how many entities and which locations should be included into the program. Some of the function can be partly centralized in a shared service and partly processed at the individual company level. Organization can have several RPA programs in each region or one global program. It is recommended that organizations start with back office functions, including both shared and local service and have global RPA program. It will allow achieving the most potential from RPA program and avoid miscommunication between different RPA initiatives. (Capgemini, Zamkow, 2017)

The second question which should be considered is a sourcing model. Depending on the dimensions determined by the scope organizations should decide how much of automation they wish to build by their own and how much external support they would need. It is strongly recommended that even with the large scope of automation to be done and involving professional firms, companies should have some internal skills to better leverage the outside help. At a minimum, organizations should have their process owners become RPA proficient, so that they can understand by themselves what should or should not be automated. (Accenture, 2016)

Another dimension of RPA program is the business case, which will be impacted by the scope, ambition, speed, and the sourcing option. While business case is necessary, it is important to have several key parameters, cultivated as the automation progress. If the business case relies on increased productivity by repurposing or reducing FTEs, two parameters have the most importance: number of FTEs detached per process automation and number of robots used per automated process. The more processes being automated, the easier optimal assignment of robots to different tasks at different times which improves the ratio robot to automated process. (Capgemini, Zamkow, 2017)

The final part of the RPA plan is an operating model. The operating model functions can be divided into three categories:

- Continuous process automation development and change management which consists of continuing to develop new RPA solutions as well as supporting modifications required in already implemented processes.
- Operational support, which provides 24/7 monitoring and troubleshooting of robots, capacity management of optimal usage of robots across processes, performance management, collaboration with IT and security departments.
- Technical support, which consists of troubleshooting any technical issue that may arise and first and second level technical support.

Any of the three main functions can be both entirely in-sourced to entirely out-sourced. The input for the assessment of process to be automated can either be fully centralized according to the determined methodology relevant to an entire organization or partly decentralized and performed by trained RPA specialists involved in the most important divisions or departments. However, it is recommended that the prioritization itself be centralized. (Moayed, 2017)

Common mistakes

Often the best practical form of guidance consists of clearly stating pitfalls that might wreck endeavor and provide some solutions to avoid them. The following list of common mistakes has proved to be the main cause of failed or difficult RPA roll-outs. Recognizing them at the beginning and avoiding them might help organizations increase their chances of a successful and large RPA roll-out. (Moayed, 2017)

The first common mistake is considering RPA only as an IT topic. Compared to traditional IT solutions RPA with its user-friendly interface is relatively easy to develop. It shifts the balance of required knowledge more into the process understanding rather than IT. This allows business people to be able to create necessary automation relatively quickly. From the other side, the agile approach of RPA development makes IT professionals uncomfortable, as they compare the ease of such kind of technology with their hard-earned technical skills. That means, organizations should not expect IT professionals to be the pushing force for the RPA roll out.

However, totally forgetting about IT could be the other side of the common pitfalls. Practically to deploy RPA there is a need for close interactions with the IT department. From the more mundane but crucial issues of RPA infrastructure set-up and access rights for robots, to more important issues such as future application roll-outs, changes and decommissioning, all of which can affect the performance of robots. While RPA is business driven, it most certainly needs to be IT governed. (Accenture, 2016)

The other danger for successful RPA implementation can come from the people who request to automate the least suitable process for RPA. Selecting the wrong process for automation and not prioritizing them methodically runs the risk of low potential process automation. It is recommended to have an accurate methodology for examining processes according their potential for productivity gain as well as their complexity and complementarity. (Capgemini, Zamkow, 2017)

The desire to automate too much of a process is also one of the common pitfalls. An attempt to cover all possible exceptions of the process may lead to complex, time-consuming workflows, which are hard to develop and maintain. It is recommended to have some steps where there will still be some human intervention.

The last typical mistake is using inappropriate delivery methodology. While traditional software delivery methods require detailed documentation and business requirements, using such kind of methodology for RPA is considered as over-engineered. It means that only relevant details which contribute to implementation quality assurance or crucial for support and maintenance should be documented. Agile delivery methodology needs to be adopted to allow developing process automation in a three to five-week cycles depending on complexity. (Moayed, 2017)

2.3 Machine learning overview

While RPA provides fast, non-intrusive way of automation, it has its limitations. As it was previously discussed, process being automated need to be well-defined according to standard operating procedures which have clear rules and parameters for decision-making. However, some of the processes require applying human judgement and expert knowledge. In such case, RPA cannot be effectively used for process automation. The example for this situation can be the problem of ticket routing described in Current

state analysis section. Applying RPA in combination with machine learning helps to overcome RPA's limitations and enhance its capabilities.

Machine learning is a subfield of artificial intelligence (AI) concerned with algorithms that allow computers to learn. By learning in this context, it means that algorithms are able to infer information about the properties of the input data and make predictions about new data which algorithms have not seen before. The capability to learn is possible because almost all data contains patterns which can be generalized by machines. The important aspects of the data are determined during the process of training machine learning model. (Segaran, 2007: 4)

Machine learning can be applied for practical problem domains in three different ways: supervised learning, unsupervised learning and reinforcement learning. The purpose of supervised learning is to build the model, which is trained from labeled data that allows to predictions for unseen data. Unsupervised learning deals with unlabeled data and explores the structure of data to extract meaningful information without the guidance of a known outcome variable. The goal of reinforcement learning is to create a system (agent), which improves its performance based on interactions with the environment. The learning process includes so-called reward signal which is given based on measurement how well the action was performed. (Sebastian, 2015: 3-7)

There are thousands of machine learning algorithms available. In order to not get lost in this huge space, it is important to understand the main components of any algorithm. These components are:

- Representation. Choosing a representation of a learner is tantamount to choosing the set of classifiers that it can possibly learn. This set is called hypothesis space.
- Evaluation. An evaluation function or scoring function is used to distinguish which algorithm is more suitable for the specified problem.
- Optimization. The optimization technique is needed to search among the algorithms for the highest-scoring one. (Domingos, 2012)

Supervised learning

As was earlier mentioned supervised learning algorithms are suitable for the problems where both inputs and outputs can be perceived. Based on training data, the algorithm

generalizes the function to be able to correctly map all possible inputs to its labels. As a result, the algorithm is expected to generate correct outputs for the inputs it has not seen before. Supervised learning can be divided into two subcategories: classification and regression. Classification tasks deals with discrete class labels, such as email spam filtering. The outcome of the regression is a continuous value, such as predicted price for the house. (Talwar and Yogesh, 2013)

The mechanism of solving supervised machine learning problem is illustrated on the figure 2.

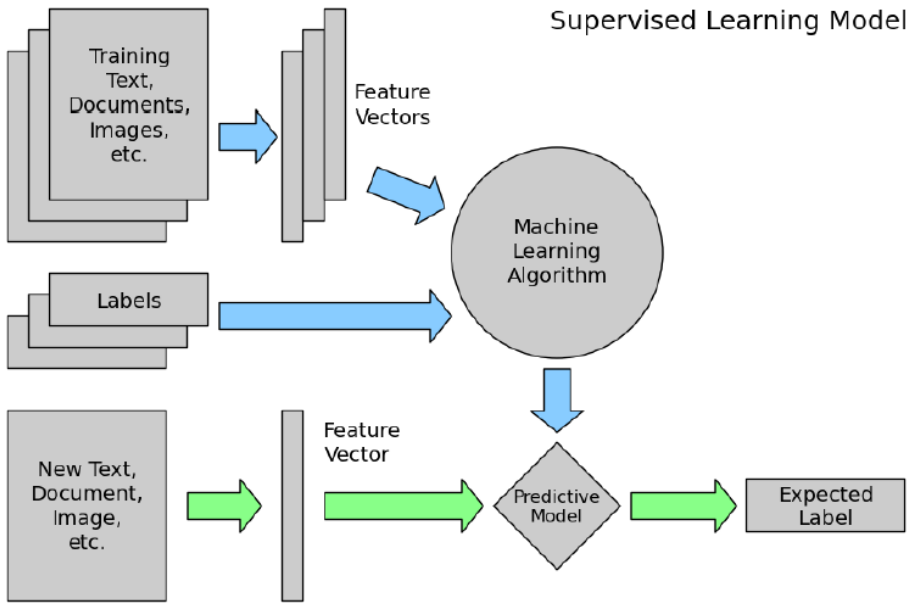

Figure 2. Working mechanism of supervised machine learning. (Talwar and Yogesh, 2013)

The steps shown on the diagram can be described in the following way:

- Determine the type of training examples.
- Gather a training set
- Determine the input feature representation of learned function.
- Determine the structure of learning function and corresponding learning algorithm.
- Complete the design and train the machine learning model on the gathered dataset.
- Evaluate the accuracy and the performance of the built function both on train and test datasets.

Two of the most common machine learning problems are overfitting and underfittimg. Overfitting occurs when the model performs well on the training dataset but does not generilize well on the new data. In case of underfitting, the model does not perfom well on both training and testing datasets. This means, the model is not complex enough to capture the pattern in the dataset. The most popular method to combat overfitting is adding regularization term to the evaluation function. This term can penalize classifiers with more structure, thereby favoring smaller ones with less room to overfit. (Sebastian, 2015: 65-66)

The most widely used supervised machine learning algorithms are support vector machines (SVM), Naïve Bayes, decision trees, random forest, logistic regression, K-nearest neighbors. They will be discussed in more details in the next subsection. Besides already mentioned algorithms, artificial neural networks and deep learning models are also used in supervissed machine learning tasks. However, such kind of models requires relatively large amount of data, more experience and time to deal with a wide range of tuning parameters.

2.4 Text classification

Text classification is associated with the assignment of a text document to a predefined set of categories, using a machine learning technique. Text classification finds its application in content management, contextual search, spam filtering, opinion mining, email sorting, sentiment analysis, etc. The classification implementation is commonly based on the basis of significant words features that are extracted from the text document. This kind of task belongs to the supervised machine learning branch, as it uses already labeled set of classes. The communication and documentation in organizations is mostly maintained in a form of textual electronic documents and e-mails. Because of information overload, efficient text categorization and retrieval of related content has achieved significant importance. (Mita and Mukesh, 2011)

The figure 3 shows the main flow of text classification process. It includes 6 main steps: document collection, text preprocessing, feature extraction and feature transformation, choosing the machine learning algorithm, its training, testing and performance measurement.

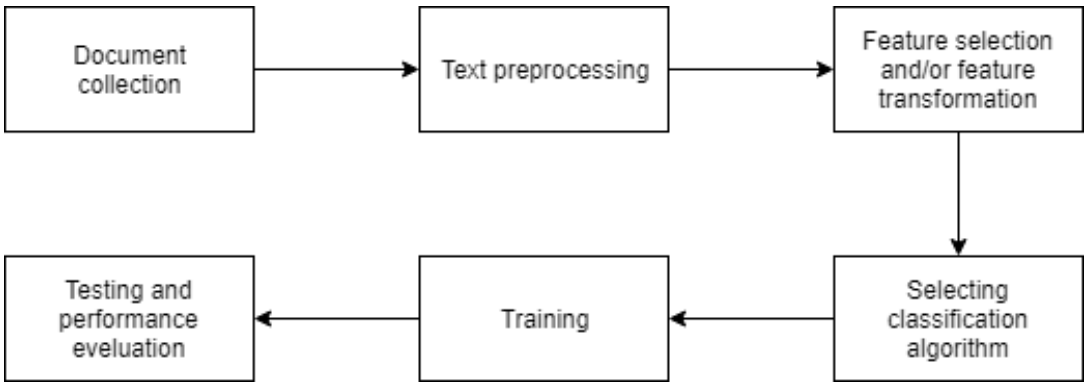

Figure 3. Main steps of text classification.

The first step of text categorization includes gathering needed amount of documents. They can be represented in different format such as html, .pdf, .doc, .xlsx, web content etc.

A document is a sequence of words, which means that each document can be represented as an array of words. The set of all the words in a training set is called vocabulary or feature set. (Ikonomakis et al. 2005) One of the main characteristics of text classification problem is the high dimensionality of the feature space. The number of unique terms in the feature space can reach thousands of words. This increases the complexity of machine learning algorithms used for classification and reduces the accuracy because of redundant or irrelevant terms in the feature space. (Foram and Vibha, 2016)

Data preprocessing phase allows significantly reduce the size of the input text documents. The goal of preprocessing is to represent each document as a feature vector. This step plays a significant role in determining the quality of the next stage. It is important to select words that carry the meaning and eliminate words that do not contribute to the documents' distinguishing. Text preprocessing incules tokenization, specific stop-word elimination, stemming and indexing. (Mita and Mukesh, 2011)

Tokenization is the process of substituting a sensitive data element with a non-sensitive equivalent, named token, which has no external or udable meaning or value. A document is considered as a string and then splitted into a list of tokens. (Upendra and Saqib, 2015) Not all of the words presented in a document are useful for training the algorithm. Such kind of words as auxilary verbs, conjunctions and articles are called stopwords. These words are frequent words that carry no information and

appears in most of the documents (‘the’, ‘of’, ‘and’, ‘to’, etc.) should be removed from the documents during the preprocessing step. (Ikonomakis et al. 2005)

One more common pre-processing technique which is used to reduce the size of the initial featuer set is called stemming. Stemming transforms words to their stem/root which incorporates a great deal of language-dependent linguistic knoweledge. Stemming is based on hypothesis that words with the same stem or root explain the same or relatively close concept in text. For example, the words “train”, “training”, “trainer” and “trains” can be replaced with “train”. One of the most popular algorithms used for stemming English words is Porter’s stemmer algorithm. Application of this algorithm allows reducing the vocabulary of the training dataset by approximately one-third of its original size. (Mita and Mukesh, 2011)

Document indexing is also one of the pre-processing steps which is used to reduce the complexity of the doucuments and make them easier to manage. Documents should be transformed from the full text version to a document vector. One of the commonly used representaions is a bag-of-words representaion. Bag-of-words model represents text as a numerical feature vector. Feature vector is constructed from each document that contains the counts of the words from the vocabulary of how often they occurs in the particular document. The unique words in each document represent only a small subset of words in the entire bag-of-word vocabulary. Because of this, the feature vector is called sparse, as it will consist of mostly zeros. (Sebastian, 2015: 236) This implies the drawback of such kind of representation: high dimensionality of the representation, loss of correlation with adjacent words and loss of semantic relationship that exists among the terms in the document. To overcome these problems feature selection and feature reduction techniques are used. (Vandana and C Namrata, 2012)

Word embedding is another text representation technique, widely used in deep learning models, such as recurrent neural networks. Word embedding produces one vector per word and uses a dense distributed representation for each word. The representation is learned based on the usage of words, which provides similar representation for similar words, compared to bag-of-words representation where each word has its own vector, regardless their usage. Although word embedding methods allows obtaining word-context relations for disambiguating document classes, it also requires huge amount of training data and can be computationally slow. (Jason Brownlee, 2017)

Feature selection is a process of selecting the subset from the original feature set on the basis of importance of features. The goal of this step is to reduce the dimensionality of the dataset, which leads to improvement of scalability, efficiency and acuracy of a text classifier. Feature selection methods utilize an evaluation function which is applied to a single word. Scoring of individual words is performed by using some of the measures, such as document frequency, term frequency, mutual information, information gain, odds ration, Chi-square, Gini index, expected cross entropy. The common aspect for all of these feature-scoring methods is that they conclude by ranking the features by their independently determined scores and then select the top scoring features. (Ikonomakis et al. 2005; Vandana and C Namrata, 2012)

One of the most popular and powerfull techniques is term frequency-inverse document frequency (TF/IDF). Since different terms have different level of importance in a document, the term weight is assosiated with every feature as a valuable indicator. Two main factors affect the importance of a term in a text: term frequency (TF) and inverse document frequency (IDF). TF is a weight of each word in a document which depends on the distribution of each word in a document. It indicates the importance of the word in the text. IDF of each term in the documents' collection is a weight which depends on the distribution of each word in the documents' collection. It indicates the importance of each term in the entire collection. TF/IDF utilizes both TF and IDF to determine the weight of a term. (Pritam et al. 2013)

Mathematically TF/IDF can be defined as the product of the TF and IDF:

$$\mathrm{TF/IDF(t,d)} = \mathrm{TF(t,d)} * \mathrm{IDF(t,d)} \quad (1)$$

TF(t,d) in the equation (1) means the number of times a term *t* occurs in a document *d*. *IDF(t,d)* inverse document frequency which is calculated according to the formula below:

$$\mathrm{IDF(t,d)} = \log \frac{\mathrm{n_d}}{1+\mathrm{DF(d,t)}} \quad (2)$$

DF(d,t) in the equation (2) represents the number of documents *d* that contains the term *t* and n_d is the total number of documents. The result of TF/IDF is a vector with various terms along with the term weight. (Sebastian, 2015: 238)

Feature transformation is another technique that is used to reduce the feature set size. It can be defined as a process of extracting a set of new features which were generated during the feature selection phase. The most common feature transformation methods are principal component analysis (PCA) and latent semantic indexing (LSI). (Foram and Vibha, 2016)

LSI technique is based on projection of the documents with latent semantic dimensions. The latent semantic space has fewer dimensions than the original space. LSI relays on the application of singular value decomposition to a matrix. (Pritam et al. 2013)

The goal of PCA method is to learn discriminative transformation matrix in order to decrease the initial feature space into a lower dimensional feature space to reduce the complexity of the classification function without affecting the accuracy level. The transform is derived from the eigenvectors corresponding. The covariance matrix of data in PCA corresponds to the document term matrix multiplied by its transpose. Entries in the covariance matrix represent co-occurring terms in the documents. Eigenvectors of this matrix corresponding to the dominant eigenvalues are now directions related to dominant combinations can be called “topics” or “semantic concepts”. A transform matrix constructed from these eigenvectors projects a document onto these “latent semantic concepts”, and the new low dimensional representation consists of the magnitudes of these projections. The eigenanalysis can be computed efficiently by a sparse variant of singular value decomposition of the document-term matrix. (Ikonomakis et al. 2005)

It is very important in PCA to determine the number of principal components. This number should be chosen among other principal components to represent data in the best way. Different types of criteria are used in order to determine the optimal number of principal components, such as broken-stick model, cross-validation, Velicier's partial correlation procedure, Kaiser's criterion, Barlett's test for equality of eigen-values, Cattell's screen-test and cumulative percentage of variance. (Foram and Vibha, 2016)

The next phase after feature selection and transformation is a selection of machine learning algorithm and its training. Many machine learning algorithms has been proposed for text classification task, which are differ in the approach: decision trees, Naïve Bayes, neural networks, support vector machines and many others. However, text classification still remains an important area of research as the effectiveness of current

classification algorithms is not faultless and needs improvement. (Ikonomakis et al. 2005)

The Naïve Bayes is a probabilistic classifier based on Bayes theorem with strong and naïve independence assumption. This is one of the most basic text classifiers, widely used in email spam detection, email filtering, language detection and sentiment analysis. Naïve Bayes is computationally efficient and it does not require a huge amount of training data. The assumption of conditional independence is breached by real-world data with highly correlated features thereby degrading its performance. (Upendra and Saqib, 2015)

Unlike Naïve Bayes, decision trees do not assume independence among its features. The relationship between the attribute in a decision tree is sorted as links. Classes are going to be rejected until the right class is reached. The purpose of a decision tree is to split the feature space into two different parts according to the specific threshold. A class is assigned with a feature vector along with a path of a decision tree. Decision trees are suitable when there is relatively less number of attributes to consider, as it becomes difficult to manage for large number of features. (Foram and Vibha, 2016; Mita and Mukesh, 2011)

SVM algorithm requires both positive and negative training data. These data is needed for the algorithm to search for the decision surface that best separates the positive and negative data in the n-dimensional space, called hyper plane. The document representatives which are closest to the decision surface are called the support vector. Compared to other classifiers SVM achieves outstanding effectiveness and performance accuracy and are capable to solve multi-label class classification. (Vandana and C Namrata, 2012)

Artificial neural networks can be a suitable approach to model complex relationships between inputs and outputs in order to find patterns in data. Neural network classifier consists of network of units, where the input represents terms and output represents the category. Each term is assigned with the weight and the activation of these units is propagated forward through the network and the value that the output unit takes up as a consequence determines the categorization decision. (Upendra and Saqib, 2015)

The last step of text classification process is testing the model and measuring performance. There are plenty of methods to measure the effectiveness, but precision, recall and accuracy are most often used. To determine these metrics, one must understand if the document was true positive (TP), true negative (TN), false positive (FP) or false negative (FN).

- TP – document was correctly classified as relating to a category.
- TN – document was correctly classified as not relating to a particular category.
- FP – document was assigned to the category which it is not related.
- FN – document was not assigned to the category but should be.

Precision (π_i) is a conditional probability that random document *d* is classified to the category c_i. It shows the ability of classifier to assign document under correct category opposed to all documents placed under that category, both correct and incorrect. Formula (3) shows the mathematical representation of precision.

$$\pi_i = \frac{TP_i}{TP_i + FP_i} \tag{3}$$

Recall is a probability that, if a random document d_x should be classified under category c_i, this decision is chosen. This metrics is calculated according the formula (4) below:

$$\rho_i = \frac{TP_i}{TP_i + FN_i} \tag{4}$$

Accuracy is one of the most common metrics to measure the efficiency of categorization technique. It represents the ration between the numbers of correct predictions (TP+TN) with the total number predictions. Accuracy values are much less reluctant to variations in the number of correct decisions than precision and recall. (Ikonomakis et al. 2005)

3 Current state analysis

This chapter explains the existing customer support system at the case company, its weaknesses and currently ongoing improvements, and highlights the areas for further enhancement.

Hundreds of tickets are passed through the Basware Issue Tracking System Management (ITSM) daily. When customers report a problem, a corresponding ticket is created in the system and serves as a token in the problem management process. Depending on the nature and complexity of the problem, tickets must be routed to the appropriate expert group, which will be handling the problem resolution process. The goal of the whole process is to resolve the ticket as quickly as possible in order to satisfy customer needs and minimize the caused disruptions.

Ticket routing plays a critical role in IT problem management. Currently, tickets are generally routed based on human decisions. Every time, experts must go through the problem description and assign the ticket to the correct group. A number of time, tickets could be mistakenly assigned to the inappropriate group, which leads to the increase of processing steps. The resolution would take longer time and could probably cause customer dissatisfaction. Figure 4 shows the schematic representation of the ticket routing flow.

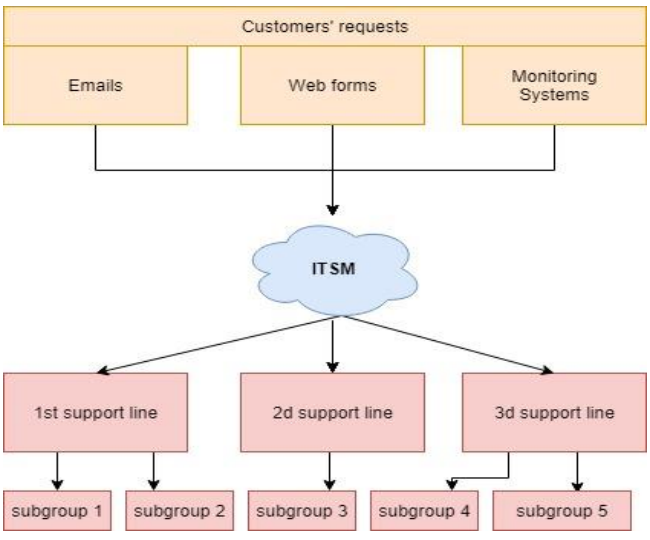

Figure 4. Customer issue ticket routing example.

As illustrated in figure 4, customer ticket is processed in ITSM system through two levels of hierarchy, where each first level line can have multiple subgroups underneath. Addressing the issue tickets to the right expert in the support team is a crucial step which affects better resource allocation and improved end user satisfaction.

In addition to tickets routing, the support team also handles number of supplementary tasks, such as adding the language code to the ticket, resolving outdated, unresponsive tickets and other tasks. All of these duties fall into the category of repeated, routine and error prone tasks, which means that automation could be an applicable in existing system.

RPA has already passed the phase “proof of concept” at the case company. Implemented automation performs language detection of tickets and assigns the corresponding language code to them. It helped to save hours of work time, weekly, and allowed employees re-direct their attention to the more meaningful tasks.

In order to further improve the existing customer support system, the automation for tickets routing is proposed for the development. Concerning the limitations of plain RPA, it should be used in symbiosis with machine learning algorithms. It would make automation intelligent and allow decision making, based on the textual content of the customer ticket.

4 Technologies and methods

This section presents the technologies that have been used during the implementation phase and describes the workflow process of the project.

4.1 Project workflow

The project implementation can be divided into five main stages: analysis, design, implementation and testing. The goal of analysis phase was to form a clear understanding of all details of the existing process and discover the opportunities for further improvements. During this step the documentation regarding the observed process was investigated and the interviews with subject-matter experts were conducted. As a result of analysis, generated idea of a solution was taken into design phase, whose purpose was to build tangible representation of the proposed system. The design plan presented the parts of the solution system, their interactions and place of integration into existing process.

The implementation phase was divided into four main sub-steps, illustrated on the figure 5.

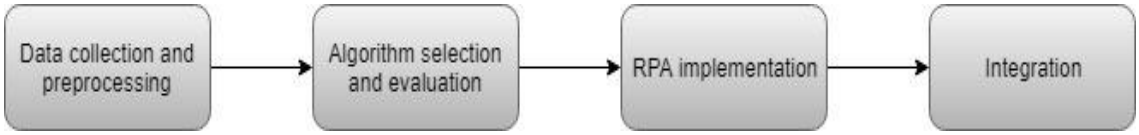

Figure 1. Implementation steps

First of all, it was decided what kind of data is required, its format and volume. After obtaining requested dataset, it had to be analyzed and preprocessed in order to give strong signals during the algorithm training. On the second step, multiple algorithms were trained with different parameter sets and evaluated along different metrics. The result of this step was a trained supervised machine learning model that was used for text classification.

RPA implementation included designing and building robotics workflow, handling possible failures and exceptions. During the integration step all implemented parts of the system were deployed in test environment and the interactions between them were

established. The final testing phase of the workflow involved gathering the required information of the deployed process in order to evaluate its robustness and performance. The results of the testing stage are supposed to be presented to the stakeholders for the further decision regarding production deployment.

4.2 Tools

From technology point of view, the current project can be divided into two parts: robotics and machine learning. Each of these parts has its own set of tools used during implementation.

Robotics platform

Many RPA platforms are currently presented on the IT market, such as Blue Prism, Automation Anywhere, UiPath, Kofax and others. Each of these platforms has its features, advantages and drawbacks. UiPath platform was chosen for this project. The choice of the platform was based on availability of online materials, ease of learning, simplicity of obtaining licensee agreement and pricing.

UiPath platform presents its product with three main components: UiPath Studio, Orchestrator and Robot. Studio is a workflow designer tool used for modeling the automation of the process by drag and drop activities, which are desired to be performed. When the actual workflow is built, it should be executed by software, which is in UiPath environment called Robot. The Robot can execute workflow based on attended or unattended automation types. In case of unattended automation, the Robot can be interpreted as a digital employee, where the workflow generated from Studio is a set of rules according to which Robot is working.

The Orchestrator is a scalable server platform which allows deploying thousands of robots. It serves as a centralize platform where robots can be scheduled, monitored and audited. In addition it provides API for integration with third party applications.

Machine learning tools

For handling data preprocessing tasks and machine learning training, Python 3.6 programming language were used in combination with its open source libraries:

- Pandas – powerful data analysis toolkit.
- Natural language toolkit (NLTK).
- Scikit-learn – machine learning library.
- Tornado – Python web framework.

Amazon web service (AWS) platform was used for deploying web server with machine learning model. All the data gathered during the automation testing were gathered and analyzed by Splunk, software platform for searching, analyzing and visualizing machine generated data.

5 Proposed solution

The current section describes implementation process of the proposed automation. It starts with design overview, data analysis and processing, continues with the selection of machine learning algorithm and modeling RPA workflow, finishes with integration and performance evaluation of the automation in test environment.

5.1 Analysis and solution design

As was previously discussed in the section 3, the current ticket routing systems at the case company has all features to be a candidate for RPA. After diving deeper into the problem, it was revealed that the most overloaded group for resolving the tickets is a 1st line support group. In order to increase the ticket processing speed and release the work force from the routine rule based duties it was decided to embed RPA robot into the process to automatically perform classification and assigning tickets which belong to the 1st line. The design of proposed solution is illustrated in figure 6.

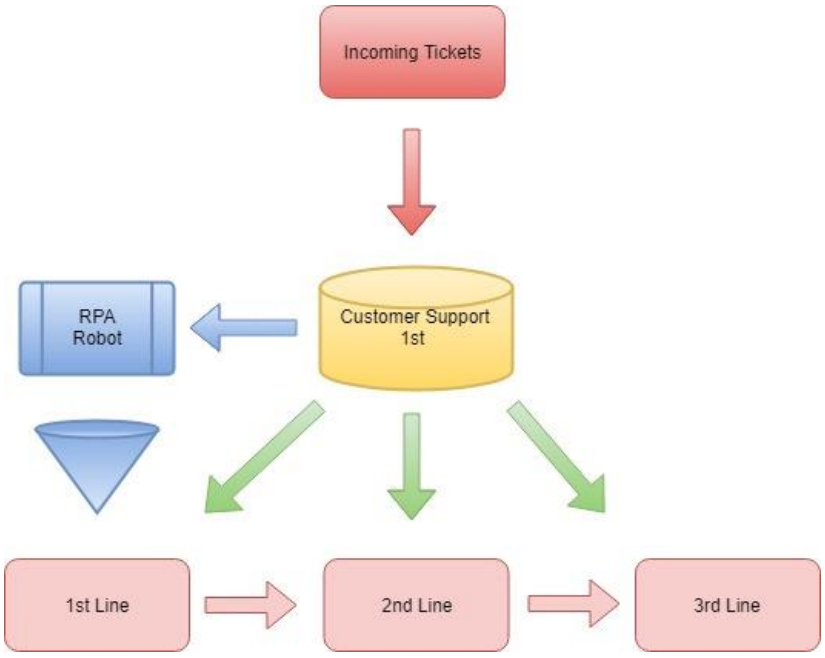

Figure 2. Design of the proposed system

Initially, all the tickets incoming from different sources are assigned to the Customer Suport 1st group. The functions of this group is validation of the information and further

routing of the tickets to the appopriate group based on the content of the problem. Tickets can be assigned directly to each of the groups and subgroups or they can be assigned from lower support line to upper line if during the processing it was revealed that the problem needs higher level of expertise. Due to the fact that 2nd and 3rd lines commonly require deeper analysis of the problem and obtaining more information it was decided that the robot will be automatically routing tickets only to the 1st line support and will leave other tickets to be analyzed by human operator.

After the ticket has been enetered into the ITSM system, the first step that human operator would perform is detecting the language of the ticket and adding the related language code. As it was previously described, this kind of task is already automated by the robotic software. The most suitable place for incorporating ticket routing would be the step after language detection as it was determined that for the current solution only the tickets with the problems described in English would be chosen for the routing step. Figure 7 illustrates this process in more detail.

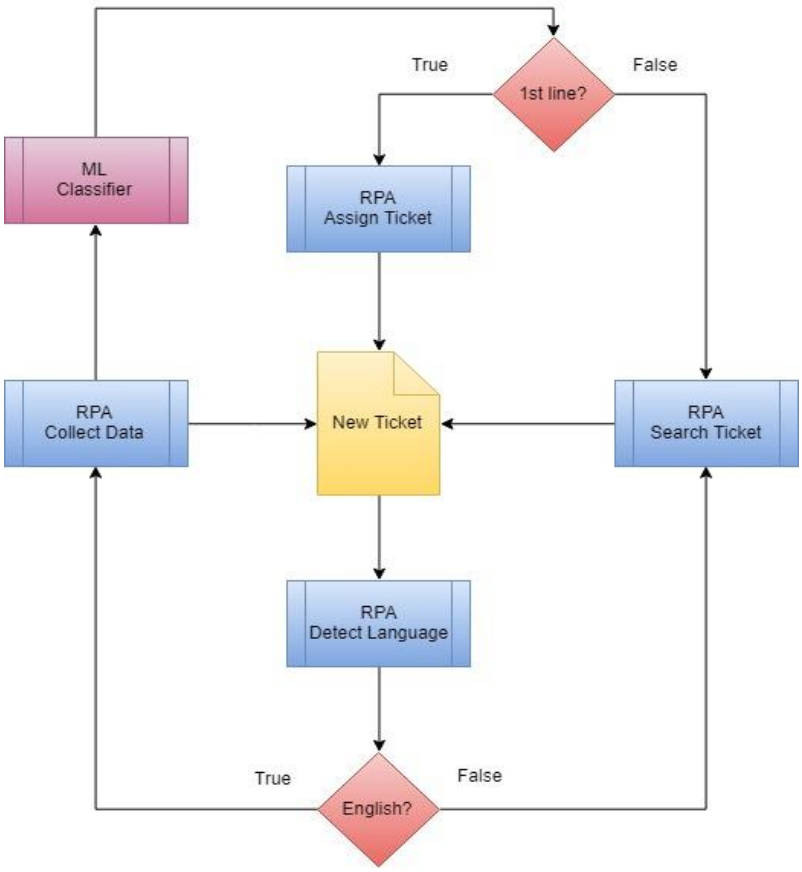

Figure 3. RPA solution design

After language detection, software robot proceeds with information retreival from the tickets whose language code is English. Then, the robot sends extracted textual data to the pre-trained machine learning model, which predicts the appropriate group for the ticket. If the ticket was classified to the 1st line support category, it would automatically be assigned to this group. In other case, no action would be performed and the ticket would be processed by human operator later and the robot continues searching for the next ticket.

5.2 Data collection and preprocessing

The first step to conduct the experience was data collection. Initially the database table containing around 200000 tickets with 280 columns was extracted. However not all the tickets were suitable for the purpose of experiment and the feature columns also had to be reduced. After interview session with experts it was revealed that the most important fields, which are taken into consideration for ticket routing, are “Service”, “Description” and “Detailed description”. The “Service” field describes the type of product the customer has an issue with. “Description” contains one sentence problem explanation and language code. This is equal to the email subject field if the issue was reported via email. “Detailed description” is actually the problem explanation, the email body text. If there were more than one email, all of them are appended to the “Detailed description” where the most recent email would be on the top. One more column of interest was “Assigned group” which is equal to the label that should be predicted. After filtering out the initial dataset by English language code, 48000 of tickets were remained.

The next step was data exploration that is the essential part of understanding the nature of the data and making decision about which features would play role in classification process and how to normalize the text. The example of the document in the dataset can be observed in figure 8.

ServiceCI	Description	Detailed Decription	Assigned Group
IP	EN FW: Something Odd?	Email From: Adoum.Namde@basware.com From: Firstname, Lastname Sent: Tuesday, April 4, 2017 13:50 To: Firstname, Lastname <example@basware.com> Subject: FW: Something Odd? Hi, Nicepak needs help with below request. Thanks, Firstname From: Firstname, Lastname [mailto: Fname.lname@company.com] Sent: Monday, April 3, 2017 3:38 PM To: Firstname, Lastname <example@basware.com<mailto:example@basware.com>> Cc: Firstname, Lastname <ex@basware.com<mailto:ex@basware.com>> Subject: Something Odd? Hey "name".....question for you. I noticed something odd with the invoices for a vendor that Sandy just today set up the monthly recurring payment for. Even though the invoice was just created on the 3-31-17 and aren't to be paid until tmrrw, they have a payment date of 3-2-17. Very odd. Can you take a look at the screenshots below.....Vendor 53232. Thanks First Invoice.....2 Screen Shots [cid:image001.jpg@01D2AE04.8522FA90] [cid:image002.jpg@01D2AE04.8522FA90] Second Invoice....2 screenshots [cid:image003.jpg@01D2AE04.8522FA90] [cid:image004.jpg@01D2AE04.8522FA90] Firstname, Lastname Accounts Payable Manager 844425.33265.1743200 ext 8241 www.example.com	CS 2nd

Figure 4. Document example

As figure 8 shows, all four columns are represented in a textual format and needed to be transformed into numeric format in order to make classification possible. The noisiest data and the most valuable are contained in the “Detailed description” column. On the example document it can be observed that there are two emails that were send for the particular issue, however it could be even more emails, depending on the complexity of the problem. It was decided to extract only the first email because the issue is initially appeared in ITSM with the description from this email and should be classified based on its content.

On the other hand, extracting only the first email led to one more problem, the language of the initial request. The issue could be started from email conversation on a local language and, after obtaining more information, the issue is submitted into ITSM system with all previous conversation and continued in English. When this problem appeared on ITSM, the English language code would be added to the ticket. In order to prevent having the dataset with documents in multiple languages, such kind of tickets should be omitted.

The final preprocessing requirement is removing all unnecessary words and characters, which does not add any value in classification process. All emails, URLs, attach-

ment names, XML code, digit-number combinations, digits and punctuation characters were decided to be removed from the dataset. This was done using Python regular expressions, as illustrated on Listing 1.

```
email_pattern = re.compile('[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}')
non_letters = re.compile('[^A-Za-z]')
url_pattern = re.compile('((http|https)\:\/\/)?[a-zA-Z0-9\.\/\?\:@\-_=#]+\.([a-zA-Z]){2,6}([a-zA-Z0-9\.\&\/\?\:@\-_=#])*')
digit_number_pattern = re.compile('[0-9%\-]+[a-z%\-]+[0-9%\-]+|[a-z%\-]+[0-9%\-]+[a-z%\-]+|[a-z%\-]+[0-9%\-]+|[0-9%\-]+[a-z%\-]+')
```

Listing 1. Defining regular expressions.

However, it was not obvious whether to apply elimination of common stop-words and stemming or not, as different researches claimed different results of this preprocessing step. Three different options: without stemming and removing stop-words, only removing stop-words, removing stop-words and stemming were tested with Naïve Bayes algorithm. The results can be observed on the figure 9.

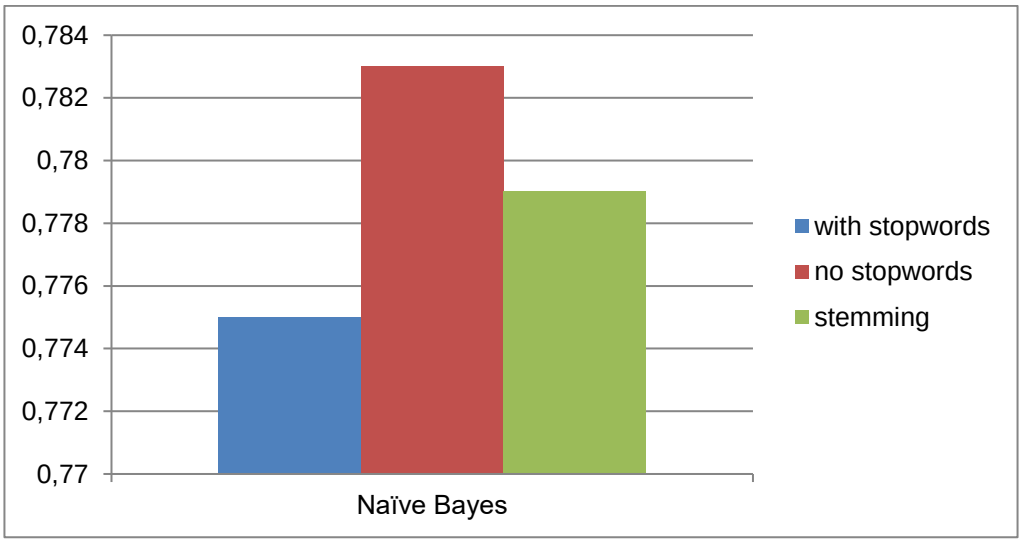

Figure 5. Accuracy performance depending on preprocessing options

The experiment was conducted with the default parameters of Naïve Bayes algorithm. Figure 9 clearly shows that the best performance was achieved in the case where common stop-words were removed and stemming was not applied. Listing 2 illustrates the steps which were taken during the preprocessing step for normalizing “Detailed description” text.

```
def normalize_detailed_description(text):
    text = text.lower()
    text = re.sub(email_pattern, ' ', text)
    text = re.sub('\[.*\]', ' ', text)
    text = re.sub('<.*?>.*?<.*?>', ' ', text)
    text = re.sub('<.*?>', ' ', text)
    text = re.sub(url_pattern, ' ', text)
    text = re.sub(digit_number_pattern, ' ', text)
    lines = text.splitlines()
    new_lines = []
    stop_flag = False
    for line in lines:
        if len(line) < 1: continue
        if 'subject:' in line:
            stop_flag = False
            del new_lines[:]
            continue
        if stop_flag == False:
            if line.startswith(lines_remove):
                continue
            if any(flag in line for flag in ending_flags):
                stop_flag = True
                continue
            new_lines.append(line)
    text = '\n'.join(new_lines)
    text = re.sub(non_letters, ' ', text)
    text = re.sub(custom_stopwords, ' ', text)
    text = ' '.join([word for word in text.split() if word not in
set(stopwords.words('english')) and len(word) > 1])
    return text
```

Listing 2. Data preprocessing for detailed description.

After making text lower case and using regular expressions for removing undesired characters the text is spitted by lines and the logic inside the “for loop” is applied to extract needed portion from the email. The condition with the “subject” keyword allows taking only the last email from the conversation. If the line starts with the word from the list “lines_remove”, this line is skipped, for example: “Email from:”, “Send to:”, “Subject”, etc. The last condition uses “ending_flags” list to check where to stop appending the lines. This list contains most common words used to close the email, such as “thanks”, “best regards”, “rgds”, “thx”. All common phrases and contraction observed during data exploration were included into “ending_flags” list. In addition, the words used for greetings at the beginning of email in different languages were also added to the list in order to eliminate the emails with languages other than English. The last step of the function removes both custom stop words and common stop words from the Pyhton “nltk” package.

After normalization, all three feature columns were merged into one and the target column with groups were converted into binary format, where the group of interest were

labeled as 1 and all other groups were labeled as 0. The last step of data preprocessing was dropping all the documents with empty values in their columns, which happened after applying preprocessing rules. Finally, the documents' corpus was reduced to the size of 33464 documents, where 17363 tickets belong to class 0 and 16101 tickets belong to class 1.

5.3 Algorithm selection and evaluation

In order to train the classification algorithm, the textual data should be transformed into numeric format. "Bag-of-words" model was used to represent text as a feature vector. However, as discussed in the section two, features in vectors can be represented in different ways: Boolean, TF, TF/IDF. All three methods were evaluated using four supervised classification algorithms: logistic regression, linear SVM, random forest and Naïve Bayes. Fivefold cross validation with accuracy score were used to measure the average performance of machine learning algorithms. In addition, it was discovered that the best performance for the current dataset was achieved using ngram range from 1 to 3. The figure 10 shows results obtained with Naïve Bayes algorithm and different feature representations.

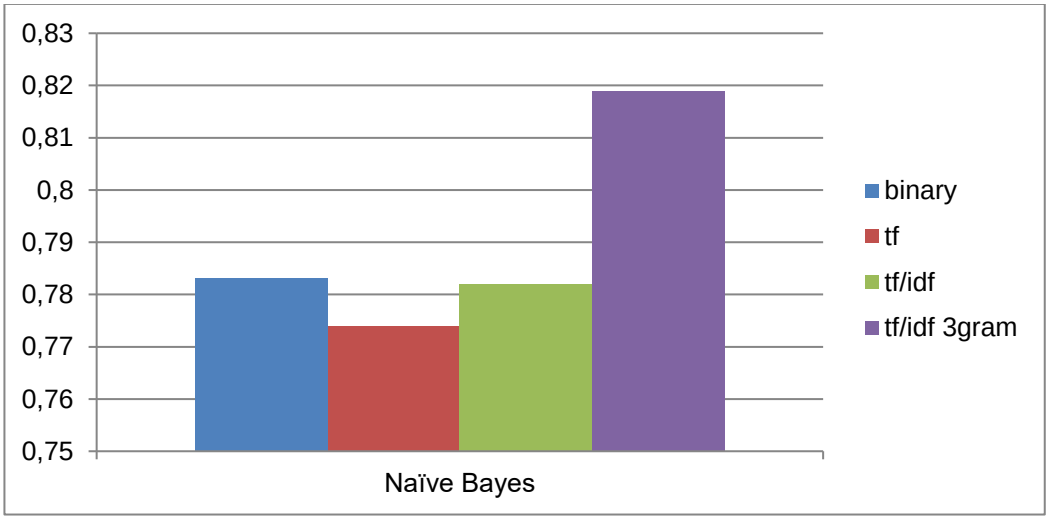

Figure 6. Accuracy score of Naive Bayes algorithm with different vector representations

It can be concluded from figure 10 that adding a three-gram range gives a significant increase in the accuracy of algorithm. All other algorithms gave the same performance, where three-gram range provided the best accuracy results.

The documents' collection was spitted into 70% training and 30% testing datasets. Grid search method was used to obtain the best tuning parameters for the machine learning algorithms. Documents' transformation into feature vectors was implemented using "TfidfVectorizer" class from the Python "sklearn" library. Main parameters that were tuned during the grid search were "ngram_range", "use_idf", "norm" and "binary". Figure 11 shows the accuracy performance of four selected classification algorithms, where "ngram_range" parameter of "TfidfVectorizer" is set to (1, 3).

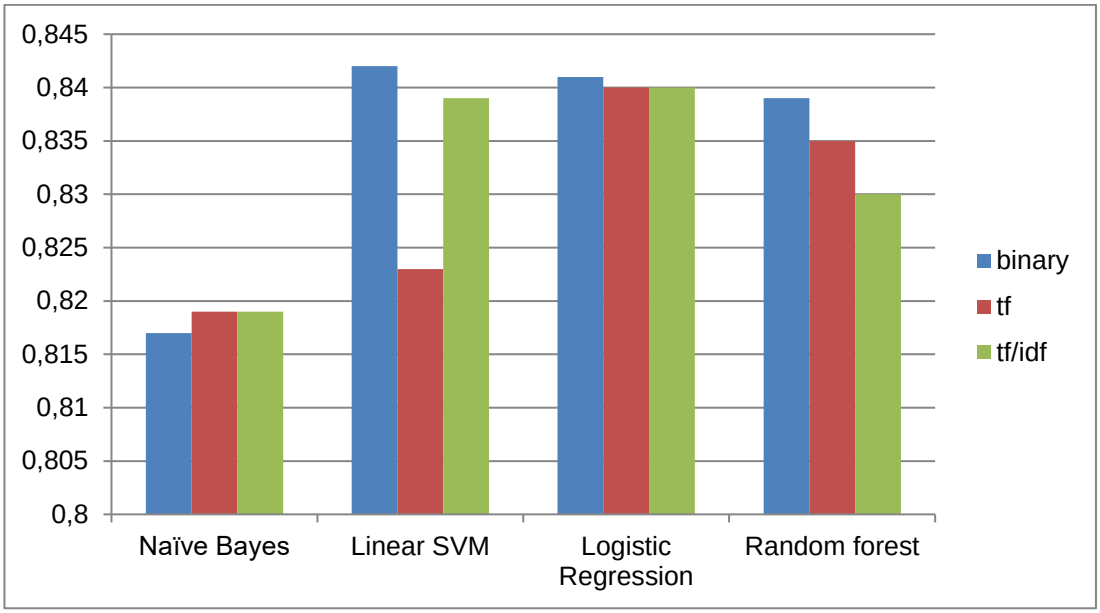

Figure 7. Accuracy performances of classification algorirhms

From the figure 11 it can be concluded that linear SVM and logistic regression provided the best accuracy scores 84.2% and 84.1% respectively. Boolean vector representation in most cases presented better results compared to other methods. Comparing the training speed of classifiers, Naïve Bayes fit the date in 0.04 seconds, SVM – 1 second, logistic regression – 6.4 seconds, while random forest training, with 50 estimators and max depth 500, took around 30 minutes.

Due to the fact that accuracy performance difference between linear SVM and logistic regression was only 0.1%, the latter one was chosen to be the production model. The reason for that was the ability of logistic regression to provide probability of belonging to the particular class, while SVM predicts the score, which is difficult to interpret and to set up a confidence interval. The use of logistic regression allowed easily establish an appopriate threshold for class affiliation prediction.

In the current project it was important to achieve not only the best accuracy score but also the highest level of precision. In case the ticket that should be assigned to the class with label 1 was not classified correctly, later human operator would do it manually. However, if the ticket should not be assigned to the group with class label 1, but by misclassification was routed to this line, it would be reassigned back to the customer support group who makes initial filtering. Such kind of mistake would increase the processing speed and add unnecessary work for human operators.

With current level of accuracy 84.1% and precision score 82%, 8.9% of tickets were incorrectly routed to the 1st line support. In order to increase the precision of predictions, probability level for the classification to the 1st line group was set as 80%. If the classifier would not be at least 80% confident that the particular ticket should be routed to the 1st line category, this ticket would be left to be classified manually by human. Such kind of probability threshold allowed increasing of the precision level up to 88,8%, while the overall accuracy of predictions was decreased to 70%. Only 4% of tickets were incorrectly classified to the class 1, which is considered as acceptable level of error for the first version.

5.4 RPA implementation

The main challenge in RPA development is creating a robust, fault tolerance and scalable architecture. First of all, it means that the workflow should be able to handle possible exception and self-recover from the error state. Secondly, the process should be divided into logical sub-processes, which can contain both system unique and reusable components. UiPath Studio allows workflow organization in two different types: sequence and flowcharts. Flowcharts are mostly used for modeling complex architectures and provide best process visualization, while sequences serve as containers for multiple ordered steps without complex conditional logic. Figure 12 presents the highest level of the implemented RPA workflow, which is called “main” workflow.

Main flow consists of four major components or sub-processes: logging into system, searching for the tickets and extracting needed information, processing tickets and signing out from the system. There are no strict rules for designing process architecture, but in current implementation process was separated into components based on reusability and logical separation into smaller tasks.

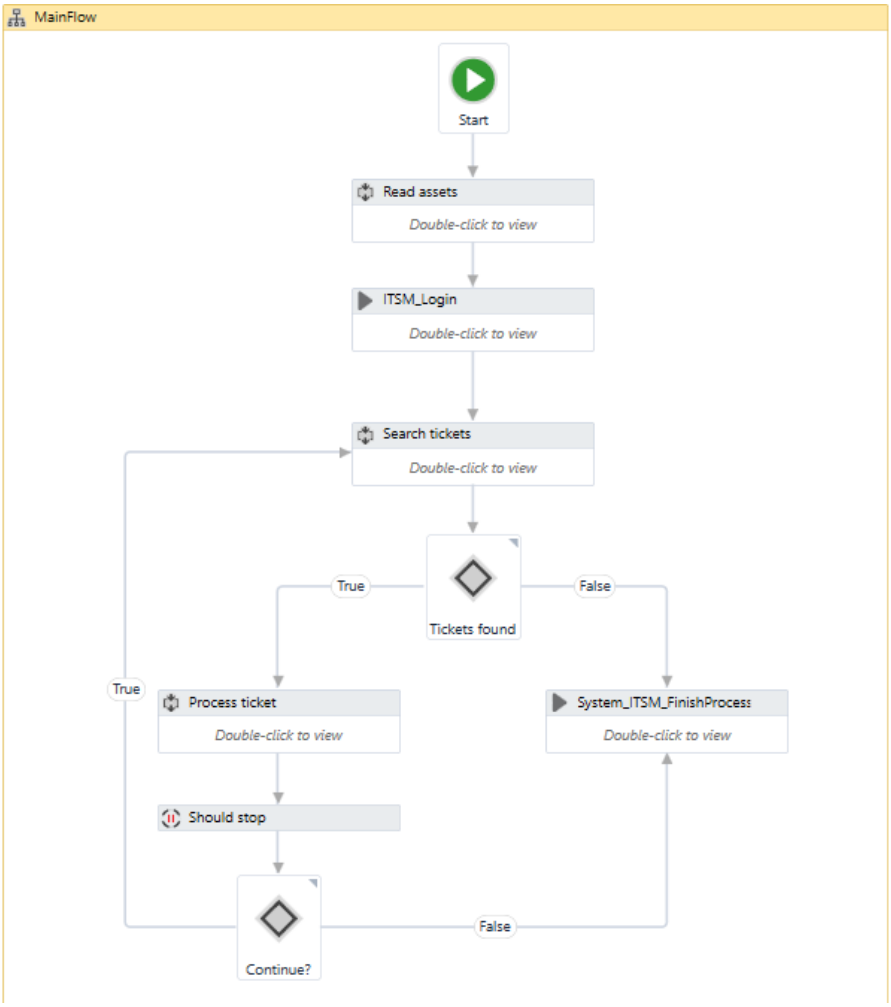

Figure 8. Main RPA workflow.

As it is illustrated on the Figure 12, “ITSM_Login” and “System_ITSM_FinishProcess” were implemented as separate workflows, because loggining into ITSM portal and signing out and closing browser can be used in other automations and the logic inside them always remains the same. “Read assets” is a complimentary sequence, which serves as a preparation step during which needed assets, such as credentials and searching string, are fetched from the Orchestrator platform. Assets can be compared to environment variables; they are stored in centralized location and can be modifyed without changing actual workflow.

The core logic of the process is contained in a loop between “Search tickets” and “Process ticket” sub-processes. The tickets are searched and processed one by one, if there would be no tickets found, the process would be complited. “Search tickets” component includes both searching ticket and extracting needed inforamtion from it,

which will be used in the next phase. This component was implemented as a sequence and part of it is represented in figure 13.

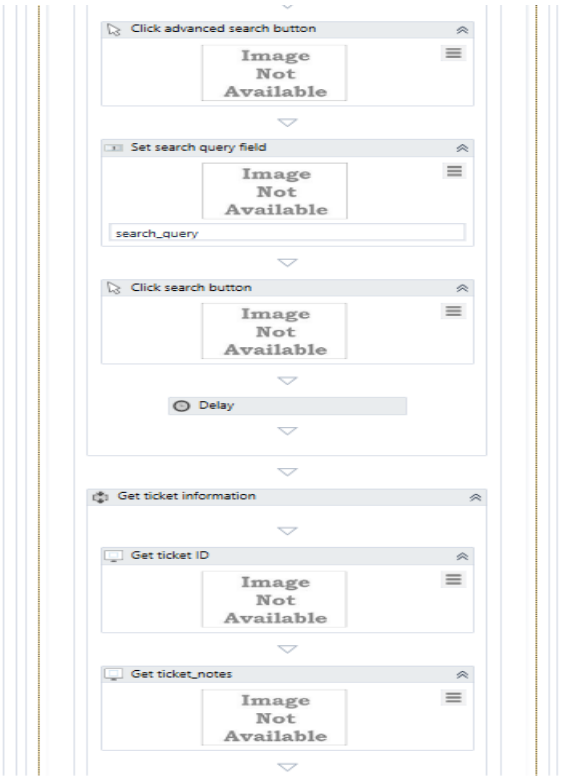

Figure 9. ”Search tickets” sequence.

The Robot, searches for the ticket using advanced search engine filled in with a searching query. After finding the ticket, it extracts information from ticket’s fields: ticket id, summary, notes and service. This information is used in the next component for detecting the language of the issue and obtaining prediction for the handling team.

The implementation of “Process ticket” component is illustrated in figure 14. At the beginning of the workflow two if-else conditions used for checking whether text exists in summary or notes field. On the next step, the first 500 characters are used as an input for the Microsoft text analytics service via REST API request for language detection. The output of the request is string with the language code, which is appended at the beginning of the summary field. All the tickets with the English language code are handled by machine learning classifier. The text from the service, summary and notes fields is sent in the body of the POST request to the web server with the deployed machine learning model, which predicts the support group for the ticket.

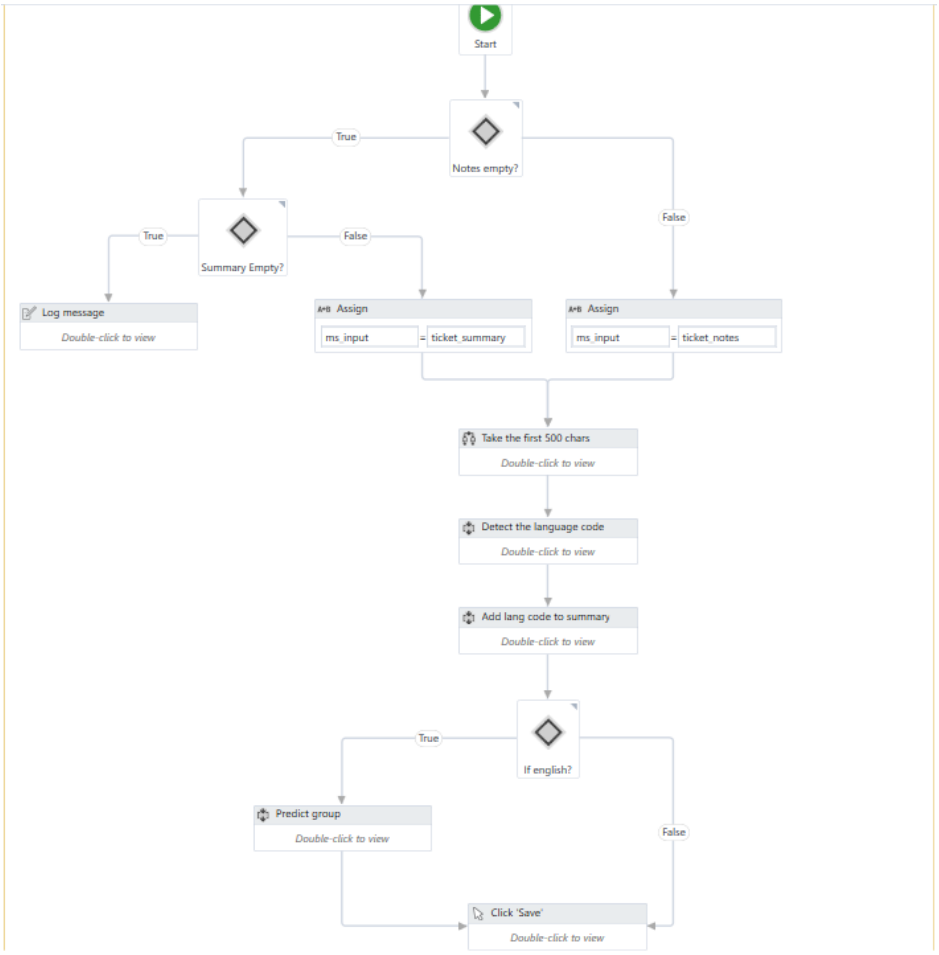

Figure 10. Process ticket workflow

All of the four components of the “main” workflow are wrapped by the try-catch block, which is used for handling exceptional situations. Exception handling was implemented in a such way that if the exception happened, the robot would start the same step from the beginning. The number of retries was set to three, which means that after the 3rd fail the exception will be thrown and the error message would be logged into monitoring system.

5.5 Integration and testing

Before running the automation in testing all of the components had to be properly configured to interact with each other. RPA automation was deployed on the Orchestrator, where was created and scheduled a process to run every five minutes on an unattended robot. A simple web server was created and deployed on AWS instance with ma-

chine learning model for classification, triggered by POST request. Robot was configured to send log messages to Splunk monitoring tool, where further analysis and evaluation were performed.

After running the automation during several days, robot was able to process 637 tickets, which were analyzed by classification performance, which is illustrated in figure 15.

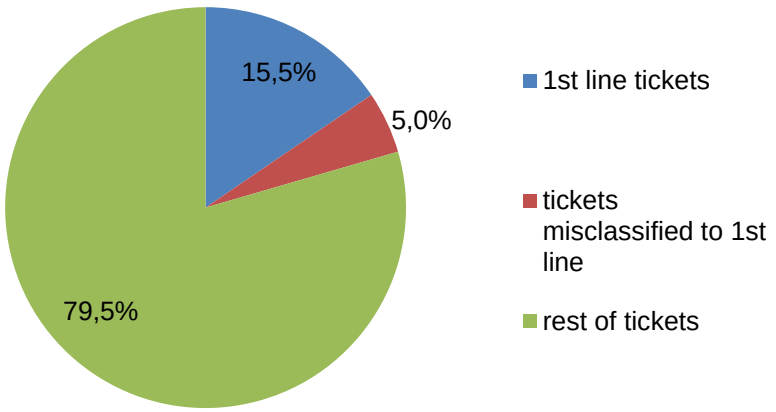

Figure 15. Classification performance

As figure 15 shows, the classifier was able to correctly route to the 1st line category 15.5% of all processed tickets, while 5% of them were incorrectly assigned to this group. The precision score of 1st line group assignment was estimated as 75.6% that is significantly lower compering to the result on test data with 88.8% precision. With 80% probability confidence threshold the accuracy score was 74.7%. The algorithm was not able to categorize to the 1st line 129 tickets, which is 20.3% from total amount.

Measuring the speed of processing tickets by robot, it was revealed that on average it took 10 seconds to process one ticket. Comparing with human, who spends around 2 minutes for the same task, such kind of results provide a significant improvement in a tickets processing cycle. However, there is a huge prospect for improving the quality of the decision making in ticket's routing.

6 Conclusion

The goal of the thesis was to discover the opportunities offered by emerging digital technologies, such as RPA and machine learning, and propose the solution for optimizing one of the Basware business processes. The implemented system provides automatic issue ticket classification and routing to the relevant support group. The automation allowed a significant increase in the tickets' processing speed, resulting in increased customer satisfaction. Also, the system facilitates the release of the human workforce from high volume and routine tasks. Human workers are able to concentrate on the processes where their professional expertise is adding value to the problem resolution.

This study proves that direct embedding of RPA into business workflows provides the ultimate flexibility in terms of triggers, logic, pre- and post-processing actions. Having access to large volumes of historical data allows application of machine learning for building decision making and prediction models. The symbiosis of RPA and machine learning evolves into intelligent process automation, which brings endless possibilities for driving savings and efficiencies achieved by powerful, sophisticated automation solutions.

The current implementation takes into account only a subset of all available data, which is based on the language. It seems that adding more language support for the classifier capabilities, essentially increases the robotics coverage of process volumes. Obtaining larger amount of data and applying more advanced machine learning methods, such as ensemble of algorithms or artificial neural networks, could shift the prediction accuracy on a higher level. Adding other customer support groups can be achieved with implementing a multi label classifier or sequential classification. However, it will add more complexity to the model and the trade-off between the accuracy and case coverage should be taken into consideration. Finally, the process itself could be expanded by including more sub-processes which now are handled by humans. These sub-processes can include data verification in the ticket and automatic attachment of the article with problem resolution. Machine learning algorithm can be trained to retrieve these articles from the companies' knowledgebase. Such kind of features would allow performing end-to-end robotic automation without human intervention in some cases.

References

Accenture (2016) 'Getting robots right' [Online] Available at: https://www.accenture.com/t00010101T000000__w__/au-en/_acnmedia/PDF-36/Accenture-16-4014-Robotic-Process-Auto-POV-FINAL.pdf (Accessed: 28 January 2018)

Asatiani, A. and Esko, P. (2016) 'Turning robotic process automation into commercial success – Case OpusCapita' *Journal of Information Technology Teaching Cases* 6 (2) pp.67-74 [Online] Available at: https://link.springer.com/article/10.1057/jittc.2016.5 (Accessed: 1 February 2018)

Capgemini, Zamkow, A. (Ed.) (2017) 'Robotic Process Automation: Gearing up for greater integration' [Online] Available at: https://www.capgemini.com/resources/robotic-process-automation-gearing-up-for-greater-integration/ (Accessed: 25 January 2018)

Domingos, P. (2012) 'A few useful things to know about machine learning' *Communications of the ACM* 55 (10) pp.78-87 [Online] Available at: https://dl.acm.org/citation.cfm?id=2347755 (Accessed: 29 January 2018)

Foram, P.S. and Vibha, P. (2016) 'Text Classification Using Machine Learning Techniques' *Wireless Communications, Signal Processing and Networking* pp.2264-2268 [Online] Available at: http://ieeexplore.ieee.org/document/7566545/ (Accessed: 10 February 2018)

Ikonomakis, M., Kotsiantis, S., Tampakas, V. (2005) 'Text Classification Using Machine Learning Techniques' *Wseas transactions on computers* 8 (4) pp.966-974 [Online] Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.9153&rep=rep1&type=pdf (Accessed: 10 February 2018)

Institute for Robotic Process Automation (2015) 'Introduction to robotic process automation' [Online] Available at: https://irpaai.com/introduction-to-robotic-process-automation-a-primer/ (Accessed: 20 January 2018)

Jason Brownlee (2017) *Machine learning mastery* Available at: https://machinelearningmastery.com/what-are-word-embeddings/ (Accessed 18 March 2018)

Lacity, M. and Willcocks, L. (2016) 'Robotic Process Automation: The next Transformation Lever for Shared Services' *The Outsourcing Unit Working Research Paper Series* [Online] Available at: http://www.umsl.edu/~lacitym/OUWP1601.pdf (Accessed: 31 January 2018)

Mita, K., D. and Mukesh, A., Z. (2011) 'Automatic Text Classification: A Technical Review' *International Journal of Computer Applications* 28 (2) pp.37-40 [Online] Available at: https://pdfs.semanticscholar.org/8077/a34c426acff10f0717c0cf0b99958fc3c5ed.pdf (Accessed: 10 February 2018)

Moayed, V. (2017) 'From pilot to full scale RPA deployment' [Online] Available at: https://cdn2.hubspot.net/hubfs/416323/Whitepapers/From%20pilot%20to%20full-scale%20RPA.pdf?t=1519561420856 (Accessed: 28 January 2018)

Ostdick Nick (2017) *Uipath* Available at: https://www.uipath.com/blog/unattended-attended-automation (Accessed 1 January 2018)

Prangnell, N. and Wright, D. (2015) 'The robots are coming, A Deloitte insight report' [Online] Available at: https://www2.deloitte.com/uk/en/pages/finance/articles/robots-coming-global-business-services.html (Accessed: 30 January 2018)

Pritam, C.G., Patil, L.H., Chaudhari, P.M. (2013) 'Preprocessing Techniques in Text Categorization' *International Journal of Computer Applications* [Online] Available at: https://pdfs.semanticscholar.org/ff34/7657082e70347a916548a9fe567ab791162a.pdf (Accessed: 15 February 2018)

Sebastian, Raschka. (2015) *Python Machine Learning*, Birmingham, UK: Packt Publishing

Segaran, Toby. (2007) *Programming Collective Intelligence*, Sebastopol, CA: O'Reilly

Slaby, J. (2012) 'Robotic Automation Emerges As a Threat to Traditional Low-Cost Outsourcing' *HfS Research* [Online] Available at: https://www.horsesforsources.com/wp-content/uploads/2016/06/RS-1210_Robotic-automation-emerges-as-a-threat-060516.pdf (Accessed: 30 January 2018)

Talwar, A. and Yogesh, K. (2013) 'Machine Learning: An artificial intelligence methodology' *International Journal of Engineering and Computer Science* 2 (12) pp.3400-3404 [Online] Available at: http://www.ijecs.in/issue/v2-i12/11%20ijecs.pdf (Accessed: 30 January 2018)

The Hackett Group (2017) 'Understanding Robotic Process Automation: Value Proposition, Deployment Model and Use Cases' [Online] Available at: https://cbps.canon.com/assets/pdf/Understanding-Robotic-Process-Automation_White-Paper_Hackett-Group_Canon-Business-Process-Services-2017.pdf (Accessed: 3 January 2018)

Upendra, S. and Saqib, H. (2015) 'Survey Paper on Document Classification and Classifiers' *International Journal of Computer Science Trends and Technology* 3 (2) pp.83-87 [Online] Available at: http://www.ijcstjournal.org/volume-3/issue-2/IJCST-V3I2P17.pdf (Accessed: 18 February 2018)

Vandana, K. and C Namrata, M. (2012) 'Text Classification and Classifiers: A Survey' *International Journal of Artificial Intelligence & Applications* 3 (2) pp.85-99 [Online] Available at: http://aircconline.com/ijaia/V3N2/3212ijaia08.pdf (Accessed: 9 February 2018)

Venkatesha, N. and Kasmera, S. (2017) White paper Robotic Process Automation (RPA) in AML and KYS [Online] Available at:

https://www.infosys.com/industries/financial-services/white-papers/Documents/robotic-process-automation-aml-kyc.pdf (Accessed: 22 January 2018)