International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol. 9, Issue.12, December 2021 E-ISSN: 2347-2693

Unsupervised Context-Based Probabilistic Text Classification

Ananya Srivastava[1*], Lavanya Gunasekar[2], Bagya Lakshmi V.[3], Navneeth Devaraj[4]

[1]Data Science Developer, Tata Consultancy Services Pvt. Ltd, Chennai, India
[2]Data Scientist, Tata Consultancy Services Pvt. Ltd, Chennai, India
[3]Principal Scientist, Tata Consultancy Services Pvt. Ltd, Chennai, India
[4]Domain Consultant, Tata Consultancy Services Pvt. Ltd, Chennai, India

[*]Corresponding Author: ananya.srivastava@tcs.com

DOI: https://doi.org/10.26438/ijcse/v9i12.914 | Available online at: www.ijcseonline.org

Received: 08/Dec/2021, Accepted: 10/Dec/2021, Published: 31/Dec/2021

Abstract — Text classification is the one of the primary tasks in Natural Language Processing (NLP). Key phrase extraction is the fundamental component that aids the mapping of documents to a set of emblematic phrases. For example, a category that includes IT documents can be described as "Information and Computer" or "Information and Technology". If a text document includes keywords such as "issue" and "order", then it belongs to "Issue Category". Multiple pre-trained and deep learning approaches are available now-a-days for semantic analysis. Word embeddings are predominant technique that provides light to find the semantic similarity between tokens/phrases using word vectors. The most widely used word embeddings are GloVe, Word2vec, BERT etc. Experimental results show that the strategy produced by this study have more precision and simplicity than that of other methods.

Keywords— Document Categorization, Keywords Extraction, Concept Learning, Multi-class Probabilistic Classification, Content Mining

I. Introduction

The artificial intelligence (AI) can improve itself through machine learning by estimating how likely two products belong to the same class in Text Classification. The practice of classification with AI is taking an increasingly substantial role in modern business.

The data analytics research field is a very vast domain with new ideas/concepts and Methodology being added to the pool periodically. So, when it comes to research and execution on the existing developments, it becomes very challenging to find the right ones which are most relevant to the requirement.

The accuracy of manual text classification can be easily influenced by human factors, such as exhaustion and expertise. It is desirable to use machine learning methods to automate the text classification procedure to yield more reliable and subjective results. Moreover, this work can also enhance the information retrieval efficiency and alleviate the problem of information overload by locating the required and relevant information.

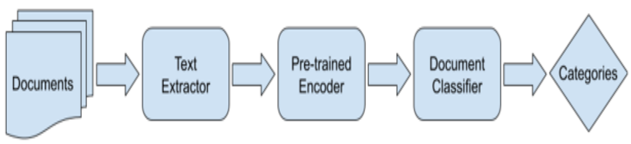

Figure 1. DLP document classification flowchart

Document Categorization is one of the standard and important NLP tasks. As illustrated in Figure 1, extract the text content from documents and use a pre-trained language model as an encoder to convert documents into numeric values. Based on the document encodings, train document classifiers in the form of fully connected neural network layers.

Since articles contain unstructured data, it is not possible to analyze articles directly through data mining techniques. Text mining provides an opportunity to apply data mining techniques by converting unstructured data points into normalized structured form. Text mining is used to extract the crux and useful information suitable for analysis and to apply machine learning models. On the other hand, data mining extracts concealed and potentially useful information from available data. It is necessary to filter, govern and classify data for people to get a quick access to information. Document classification refers to the assignment of texts to pre-determined categories. Prior to computer systems, classification was done manually. This process is not only slow and expensive but also inconsistent. That the processes are done via machine to decreases these disadvantages to a great extent. In text classification studies, it is seen that preprocessing, feature selection methods, term weighting and classification algorithms are taken into consideration.

II. Related Work

This section provides overview on the existing methodologies related to keyword extraction. Also emphasizing on the use of rule-based and word embedding technique for text classification. There are mainly three text classification approaches-

- Machine System
- Hybrid System.
- Context-based with Probabilistic System

Classification is the process of assigning documents to predefined classes. In the **context-based probabilistic approach**, texts are separated into an organized group using a set of hand-crafted linguistic rules. **ML based classifier** learns to make a classification based on past observation from the data sets. It collects the classification strategy from the previous inputs and learns continuously. Machine-based classifier uses a bag of words (BOW) for feature extraction.

The third approach to text categorization is the **Hybrid Approach**. Hybrid approach usage combines a statistical rule based and Machine Learning based approach. This approach utilizes rule-based system to create a tag and applies machine learning to train the system and create a model. Then the machine-based rule list is compared with the rule-based list. If something does not match on the tags, human intervention improves the list manually. It is the best method to implement document classification.

In this study, we used semantic analysis - context-based approach for information rich token extraction. After that, we categorize the tokens using domain guided multi-class probability estimation method. Here, the corpus has low quality data, at some instances minimal words, incomplete that doesn't make sense. So, cleaning and pre-processing becomes real challenge with insufficient information.

We have a short text description of customer call and the detailed text description. It becomes a challenge for non-expert to the understand the text description provided in this dataset. Added to that, it was lengthy with lot of noises. Therefore, for keyword extraction we used short text description.

In this case, few keywords were provided by the customer. But when we tried to map those description based on the given keywords to their respective call drivers, the results were not appropriate. The suggested keywords appeared to be insufficient.

So, the main objective of this analysis is to extract meaningful and maximum number of keywords from the short description and populate them to their respective call drivers or categories appropriately. For research, we used word2vec and rule-based methodology to get tokens in vector representation and generated analogous matches.

The very first step to collect and the data. The most important and the demanding step would be to determine the word vectorization and the classifiers. According to the results of the previous works, using content-based gives the best results when compared to other methods like transformers, flair and word2vec etc., While applying word2vec it is providing only top 'k' similar words where selection of 'k' value is not possible for all scenarios. Hence, we applied context-based with probabilistic approach.

Also, while implementing hybrid technique combined with context-based probabilistic approach, we observed the following advantages over hybrid techniques:

- The context-based probabilistic approach is a highly reliable engine which provides application with higher accuracy even the data quality is less.
- It is a condensed representation of contextual knowledge that can be leveraged across multiple industries such as retail/e-commerce, hardware, healthcare, industry supply etc.,
- The information extraction techniques have been used effectively in commercial systems which is more flexible for inserting new classes or features on futuristic data. It can be easily understood and controlled at any time.
- Easy to implement and more interpretable that reduced the human effort and domain knowledge required for classifier distribution.
- Results are more oriented towards human thought process, which makes this approach easy for developers to implement as compared to other complex hybrid techniques like word2vec, BERT etc.

III. METHODOLOGY

One should select and aggregate data labellers or classifiers to perform large-scale dataset creation or classification. Here, we will be discussing in detail about the steps involved around in the text classification and the methods we implemented and what are the challenges we encounter during their implementation.

A. Datasets

In this analysis, the American retailer hardware dataset has been used that is completely unlabelled. It consists of two parts 1) Customer Ticket and 2) Retailer Ticket. In customer dataset, total it holds 19109 rows and retailer holds 292411 rows. Customer and Retailer adopts 9 categories to categorize description. The categories provided are Consumer Privacy Request, Order Cancellation, Statement Payment or Refund due today, Order due today etc. The features are *Short Description, Description, Incident ID, Created Date Time, Closed Date Time* etc. Majorly the focus is on *Short Description, Text Description and Call Drivers*. Few keywords were already given by the client for understanding each call drivers.

Information regarding the datasets used within the research is given in Table I and Table II that shows the *Call Drivers* and *Sub – Call Drivers*.

Table 1. Excel Sheet of the Dataset Attributes

Short Description	Description	Owned By	Category	Subcategory	Status	Priority
wrong rewards card number	Rewards member numbe	Erica Palumbo			Closed	2
tool buy back questions	how to send back the bu	Christopher Hedley			Closed	2
Store Complaint ---- Auto-Closed	Was "assaulted" and "fal	Kasaundra Kania			Closed	2
No Hotsheets or Invoices downloaded or was a	No Hotsheets or Invoices	Tammy Campbell	1.7 Order En	Create Incident	Closed	3
ordered airbed opened it last night and its infl	opened air mattress- hol	Erica Palumbo			Closed	3
shipping issue ---- Auto-Closed	13365349\r\n \r\ncustom	Brittanie Ortiz			Closed	2
used reward in cancelled online order	I just got something from	Erica Palumbo	14.1 Custom	Create Incident	Closed	3
Lost order	Customer order is lost 13	Keteline Edmond			Closed	2
warehouse claim	I submitted a claim for th	Paula Sandrik			Closed	2

Table 2. Call Drivers, Sub Call Drivers and Keywords.

Call Drivers	Sub Call Drivers	Keywords
Order due today	Where is my delivery?	delivery, tracking, order, truck
	I am missing items from my delivery.	shortage, claim, damage
	I did not receive my ASN.	ASN, ERP, electronic invoice
Statement Payment or Refund due today	I am not able to view my statement on ACENET	statement, ACENET
	I need a partial credit on my statement.	Credit
	How do I make a partial payment on my stateme	partail payment
	I am missing a credit on my statement.	missing credit
	Why am I getting a credit on my statement?	credit, reset credit, Discovery credit

Pre-processing is the first step to convert unstructured data into structured form. One of the most important steps of text mining studies, may change depending on the document language, type, and the source it is obtained. However, to generate a required input, the following pre-processing steps are carried out; tokenization, stemming and stop-words filtering.

There are four sub steps involved:

1) Regular expressions are used to remove undesired characters and words from description. Examples include hyperlinks, webpages, words starting with '@' and '&', numeric values, and special characters.

2) Secondly, a list of stop words is created including auxiliary verbs, prepositions, and pronouns that are known to carry little or no semantic information. **'stopwords'** is a list of words that do not add much meaning to the sentence (e.g., 'a', 'but'). **'word tokenize'** splits up a sentence into its tokens i.e. words and punctuations whereas **'sent_tokenize'** splits up a paragraph into its respective sentences. Note that this is an experimental list to study the effect of excluding stop words on text classification.

3) Thirdly, single character from description is excluded.

4) Unicode Transformation Format (UTF) of individual emojis is used in place of the Unicode character.

Part of Speech tagging (i.e. **POS** tagging) is the process of labelling each word in a sentence with its appropriate part of speech. The **POS** tagger in python takes a list of words or sentences as input and outputs a list of tuples where each tuple is of the form **(word, tag)** where the tag indicates the part of speech associated with that word e.g. proper noun, verb, etc. spaCy library is used for getting the Named Entity Recognition (NER) and pos-tags "NOUN", "VERB", "ADJ", "ADV". Also, inflation is executed to replace the plural values with singular value, duplicate words removal, category name redundancy removal in feature/tokens. Then word frequency dictionary is developed against the extracted tokens.

Next step is to generate bigram and trigram. Bigrams are two words frequently occurring together in the document. Some examples in this corpus are: 'best practice', 'forgot send' etc., Trigrams are three words frequently occurring. Now the probability is assigned to the occurrence of an N-gram or the chances of a word occurring next in a sequence of words Why? First of all, it can help in deciding which N-grams can be chunked together to form single entities.

It also helps to make next word predictions. Say we have the partial sentence "Not good so returning". Then it is more likely to be "order" or "delivery" or "product".

It also helps to make spelling error corrections. For instance, the sentence "isue status" could be corrected to "issue status" if we knew that the word "issue" had a high probability of occurrence before the word "status" and also the overlap of letters between "issue" and "isue" is high.
Generally, the bigram model works well and it may not be necessary to use trigram models or higher N-gram models.
So, in this case, bigram is sufficient and providing better outputs.

B. Context-Based Keyword Extraction:

Contextual meaning of words is a major aspect everyone must consider while developing natural language processing models. It will be challenging to develop a language model that gives fair results to users and follow intent of the phrase.

It is very important for any NLP model to understand the semantic and syntactic meaning of words in any language. In this section experimental elaboration is made on how it can be combined and get an impressive outcome using context-based and Multi-Class Probabilistic methodology and how it differs from hybrid techniques like word embeddings.

Word embedding process is implemented using word2vec from the library genism. The pre-processed data is used as input for the word embedding process. This process uses pre-processed data in the form of tokenized text as input and produces a model output.

The results obtained are *helpful, very, informative, not bad, really, beneficial, help, easy, useful*, and *good*. These are some explanations of parameters inside word2vec and the values that is assigned to this model.

Table 3. Parameters and their description and values

Parameter	Description	Values
min_count	Minimum word count	1
size	Number of vector dimensions	200
workers	Number of parallel threads	5
window	Context word window size	5
iter	Number of training iterations	30

The major process is to extract keywords that are statistically appropriate for the provided document, also to rightly identify meaningful phrases that are most semantically related to the main theme of the document. For example, the word used is "lost" and the model shows relatively similar and meaningful phrases to the word "lost" by using word2vec.

The purpose behind keyword extraction for the top-k keywords that is selected are:

• Coherent - The ranked keywords should be comprehensible to readers, which is made possible by selecting grammatically precise and meaningful phrases that possess the characteristic of high readability by humans. For example, the phrase scientific articles have more logical than the individual words scientific and articles.

• Relevancy - The top-k keywords should be linguistically correlated to the central concept of the document.

• Good Coverage - The extracted keywords should cover all the major topics discussed in the document.

However, the outcomes provided with word2vec seems very minimal and irrelevant according to the requirement. Those results were insufficient and time-consuming for optimization.

word2vec has the inability to handle unknown or out-of-vocabulary (OOV) words in corpus. Embeddings represent each token as independent vector representation; so multiple vectors are produced for morphologically similar tokens. This becomes challenging task where internal structure of tokens is not manipulated.

However, when we searched for similar keywords using word2vec model, the results induced are insufficient, i.e, the model produced only top-k words. Our objective is to fetch maximum keyword for each *Call Drivers* and map the text phrase correctly. Nevertheless, word embedding is one the most common method that is used for keyword extraction and text categorization but depending on the dataset we need to check the model accuracy and select the relevant model accordingly.

Few examples for keyword extraction using word2vec are given below:

```
wl = ['issue']
model.wv.most_similar(positive=wl)
```

```
[('stick', 0.6267663240432739),
 ('issueid', 0.6204144954681396),
 ('external', 0.6187551021575928),
 ('statu', 0.6136037111282349),
 ('acehardware', 0.611139178276062),
 ('whole', 0.6054260730743408),
 ('successful', 0.6009618043899536),
 ('adjust', 0.583467960357666),
 ('bucket', 0.5797982811927795),
 ('lot', 0.5774184465408325)]
```

Figure 2. Some examples of word2vec outputs

C. Keyword Extraction using Rule-based Technique :

One of the most common NLP tasks is to search if a string contains a certain pattern or not. A R**egular Expression (RE)** in a programming language is a special text string used for describing a search pattern. It is extremely useful for extracting keywords. For instance, a python regular expression could tell a program to search for specific text from the string and then to print out the result accordingly. Expression can include

- Text matching
- Repetition
- Branching
- Pattern-composition etc.

The python *RegEx* match method checks for a counterpart only at the beginning of the string.

As this dataset holds limited keywords and even embeddings are propagating very minimal results, initially decided to use re.match(). So "Short Description" is converted into string and then on top of it re.match() is applied to find the respective keywords.

Below are few examples of using regular expression re.compile and re.match to extract keywords.

```
r = re.compile(".*claim")
newlist = list(set(filter(r.match, l)))# Read Note
print(newlist)
```

```
['usaa claim', 'claim store', 'claim never', 'claim damage', 'claim item', 'claim mail', 'agent claim', 'claim accident', 'file claim', 'coupon disclaimer', 'purchase nclaime', 'nclaim fnm', 'order claim', 'want claim', 'claim auto', 'nfedex claim', 'bass ett claim', 'lose claim', 'claim chagre', 'claim nmuffy', 'claim order', 'claim incorrect', 'unclaimed property', 'reject clai m', 'claim email', 'find claim', 'fill claim', 'nclaime auto', 'receive claim', 'accident claim', 'claim lose', 'quality clai m', 'customer claim', 'claim charge', 'warranty claim', 'claim receive', 'claim scraper', 'fraud claim', 'claim number', 'claim nclaim', 'report claim', 'claim birthday', 'claim form']
```

```
r = re.compile(".*replacement")
newlist = list(set(filter(r.match, l)))# Read Note
print(newlist)
```

```
['replacement white', 'replacement bulb', 'replacement ship', 'want replacement', 'card replacement', 'replacement auto', 'repl acement expire', 'replacement float', 'call replacement', 'replacement vehicle', 'replacement lose', 'issue replacement', 'repl acement part', 'request replacement', 'look replacement', 'state replacement', 'seek replacement', 'replacement request', 'upse t replacement', 'replacement credit', 'replacement seat', 'send replacement', 'item nreplacement', 'replacement reward', 'repla cement care', 'nreplacement request', 'appeasement nreplacement', 'replacement nlost', 'need replacement', 'question replacemen t', 'replacement receive', 'replacement order', 'reship replacement', 'replacement item', 'replacement card', 'replacement ca p', 'replacement wrench', 'nreplacement gift', 'replacement gift']
```

Figure 3. Some examples of Regex outputs

In simple terms, a regular expression can be compiled into a regex object to **look for occurrences of the same pattern inside various target strings** without rewriting it. Some advantages of using Regular Expression are:

- **Helps in searching specific string pattern and extracting matching results in a flexible manner.**
- **Time spent on data collection and extraction of information can be saved significantly.**

Note: List can contain bigram_list or trigram_list which is derived from text pre-processing.

D. Document Categorization using Unsupervised context based with multi -class ***probabilistic classification*** *:*

Probabilistic classification is the task to split the document set into distinct highly relative classes or groups based on nature of the document. It divides the pre-processed document set into 'k' different groups or classes by 'k' number of pre-determined Keyword Text Pattern Models through probability technique but in this dataset, there are pre-defined classifiers and based on pre-processed text it is required to map those text to their respective call drivers (classifiers) using extracted keywords.

In the last stage, text / keyword mapping is done using the multi-class probability techniques.
Initially noun phrase extraction is required so the text description is converted into list and then applied the user defined function for noun phrase extraction parser to the list of text description.

So, now a new attribute "*construct*" is formed which is having noun phrase for each of the text description.
Now comes the major challenge to map the text description based on the extracted constructs to their corresponding call drivers using the probability technique. So user defined function is constructed in which there are different levels of lists {l1, l2, l3,….,ln}. Each list contains the extracted keywords. It works as a frequency occurrence where the constructs are compared against the chances of keywords. If the frequency of keyword 'x' is high in 'l1' then it will be mapped to l1's defined *Call Driver*.

Let us consider l1 = {'lost', 'missing', 'damage', 'defective'} and l2 = {'status', 'track', 'refund', 'tax', 'exemption'} and their respective drivers would be l1 = {"lost/damage order'} and l2 = {"Order/Refund status"}

The probability of each token in the list against the contextual keywords decides the classifier. Higher the probabilities of keywords signify more chances towards the respective *Call Driver*. In above example, it will map to the {"lost/damage order'"}.

Formally, an "ordinary" classifier is some rule, or function, that assigns to a sample x a class label ŷ:

$$\hat{y} = f(x)$$

The samples come from some set X (e.g., the set of all documents, or the set of all images), while the class labels form a finite set Y defined prior to training.
Probabilistic classifiers generalize this notion of classifiers: instead of functions, they are conditional distributions Pr(Y|X), meaning that for a given $x \in X$ they assign probabilities to all $y \in Y$ (and these probabilities sum to one). "Hard" classification can then be done using the optimal decision rule

$$\hat{y} = \arg\max_y \Pr(Y=y \backslash X)$$

In more intuitive way, the predicted class is that which has the highest probability.

IV. Results and Discussion

This study proposes a novel filter based probabilistic multi-class classification method best suits for dealing huge unsupervised data. The proposed method is compared with well-known hybrid and state-of-the-art algorithms including neural networks, GloVe, ELMo, word2vec and doc2vec.

Short Description	noun_construct	main_cat
never received any of it. Damaged in transit\...	['never', 'receive', 'damage', 'transit', 'nin...	Order due today
AP-Rec - Vendor Disputing a Quantity Chargebac...	['vendor', 'chargebackocq']	All Other Requests
misfilled order customer is sending pics for r...	['misfille', 'order', 'customer', 'reship', 'm...	Order due today
order cancellation	['order', 'cancellation']	Order Cancellation

Figure 5. Output with mapped Categories

Performance indicators are very useful when the aim is to evaluate and compare different classification models or Hybrid learning techniques.

Everyone is interested in measuring the accuracy of the final classification results, i.e., the correctness of assigning categories to a set of documents. In simple words, consider choosing a random unit and predict its class, Accuracy is the probability that the model prediction is correct. In the output we have seen higher the probability of keyword of one class gives the higher chance that the document belongs to this class and when human cross-validation was done the results seems to be more accurate for Context-based Probabilistic method as compared with other Approaches.

Accuracy returns an overall measure of how much the model is correctly predicting on the entire set of data.
Therefore, Accuracy is most suited when mostly cared about Context-based Probabilistic method instead of Other Approaches.

On the other hand, the metric is very intuitive and easy to understand. Both in binary cases and multi-class cases the Accuracy assumes values between 0 and 1, while the quantity missing to reach 1 is called Misclassification Rate.

V. Conclusion and Future Scope

Performance Improvement:
Word embeddings along with regular expression and 'N-gram' analysis will maximize the keyword coverage for each classifier/*Call Drivers.* Less functional knowledge is sufficient to run the ensemble technique against any type of data. This will solve the issue where data collection and data quality are major challenges for execution.

Efficient & Adaptable:
One of the major advantages of probabilistic models is that they provide an idea about the uncertainty associated with predictions. In other words, it provides an idea of how confident the approach for its forecasting. Our unique noun phrase chunker is flexible to implement across any kind of industry data that will work best for the semantic analysis and issue area identification from the provided text. So self-optimized with less information, lower cost, fundamental or minimum infrastructure requirement with better process standardization.

It may be possible to obtain further reduction of the model size for robust and scalability. While more number of classes and features are introduced in the data, then the model becomes ambigiouis. Multiple probability will be created with almost same weightage and fails to predict the correct classifier. In future, we will investigate these shorcomings for modified/new strategy with state-of-the-art neural nets.

REFERENCES

[1] An Jiyuan and Chen, Yi-Ping Phoebe 2005, "Keyword text extraction for text categorization", in Proceedings of the 2005 International Conference on Active Media Technology, **pp.556-56 June 2005**

[2] Bhumika, Prof Sehra SS, Prof Nayyar A, "A review paper on algorithms used for text classification," International Journal of Application or Innovation in Engineering & Management (IJAIEM), **vol. 2, Issue 3, March 2013.**

[3] R Shrihari C, Desai A, "A review on knowledge discovery using text classification techniques in text mining," International Journal of Computer Applications (0975 – 8887), **vol. 111 – No 6, February 2015.**

[4] Dang, S., & Ahmad, P.H ,"A Review of Text Mining Techniques Associated with Various Application Areas", International Journal of Science and Research (IJSR), **Vol.4, No.2, pp.2461-2466,2015.**

[5] Mehmet Fatih KARACA and Safak BAYIR "Examining the Impact of Feature Selection Methods on Text Classification" Published by : (IJACSA) International Journal of Advanced Computer Science and Applications, **Vol. 8, No. 12 , 2017.**

[6] R. Janani1, S. Vijayarani "Text Classification: A Comparative Analysis of Word Embedding Algorithms", International Journal of Computer Sciences and Engineering, **Vol.7, Issue-4, April 2019.**

[7] Riya Tyagi "Review of Extraction and Classification of Key-Phrases in Scientific Publications using CRF and WEDP" Published by : International Journal of Engineering Research & Technology, Volume 09, Paper ID : IJERTV9IS060480, **2020.**

[8] Rafly Indra and Abba Suganda Girsang "Classification of User Comment Using Word2vec and Deep Learning" Published by : Advances in Science, Technology and Engineering Systems Journal, **Volume 9 No.1 , Page 643– 648 , 2021.**

[9] S. Sreedhar, S. Ahmed, P. Flora, LS Hemanth, J. Aishwarya, R. Naik "An Improved Approach of Unstructured Text Document Classification Using Predetermined Text Model and Probability Technique",Published by : International Journal of Recent Technology and Engineering, ISSN: 2277-3878, **Volume-8 Issue-2S10. January 2021.**

AUTHORS PROFILE

Ms Ananya Srivastava pursued Bachelor of Computer Application from Guru Gobind Singh Indraprastha University, Government University, New Delhi. She has also completed one year executive PG program in Data Science integrated with AI and ML from Gurgaon. She holds 3 years of experience in Tata Consultancy Services and currently employed in TCS R&D as a Data Science Developer and Researcher. Her main work comprises of Research and Develpoment in Artificial Intelligence, Deep Learning and Machine Learning techniques predominantly working in Natural Language Processing and Recommendation Engines.

Mrs BagyaLakhsmi.V has been working as a Principal Scientist at Tata Consultancy Services, Chennai for over 15 years. She has completed her B.Tech Degree in Electrical and Electronics Engineering from Government College of Technology, Anna University, Coimbatore. She also has a Post Graduate from Robotics allied to her UG. She is a pronounced Data Science enthusiast with many research activities and international publications in the field of Artificial Intelligence to her credit.

Ms Lavanya Gunasekar has been working as Data Scientist and Machine Learning Professional for past 9 years in Tata Consultancy Services. She has worked in multiple areas such as Data Management, Process Mining, Data Modelling, Data Visulations, Data Integration, Aritificial Intelligence. She has developed and offered guidance in Natural Language Processing based product prototypes for solving complex business challenges across various industries. She has completed Bachelor of Engineering in the field of Electrical and Electronics from Government Engineering College, Anna University, Erode in the year 2012.

Mr Navneeth Arjun Devaraj has been working as a Functional Consultant in Retail/E-commrce at Tata Consultancy Services with an overall experience of 20 years. He acquired an Engineering Degree in Computer Science from MGR Engineering College, Chennai. He is a 'Busimess Analytics' practitioner with expertise in Business Process Transformation, Process Re-engineering and Change Management.