



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

A Thesis

For the Degree of Doctor of Philosophy

**A Hybrid Approach for Topic Discovery and  
Recommendations based on Topic Modeling and Deep  
Learning**

Wafa Shafqat

Department of Computer Engineering

Graduate School

Jeju National University

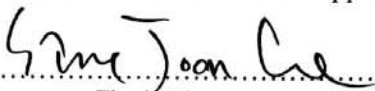
December 2019

# A Hybrid Approach for Topic Discovery and Recommendations based on Topic Modeling and Deep Learning

Wafa Shafqat  
(Supervised by Professor Yung-Cheol Byun)

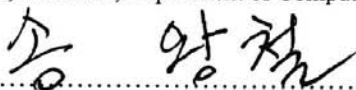
A thesis submitted in partial fulfillment of the requirement for the degree of Doctor  
of Philosophy in Computer Engineering  
2019. 12. 20

This thesis has been examined and approved.

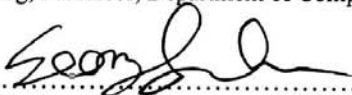


Thesis Director

Sang-Joon Lee, Professor, Department of Computer Engineering



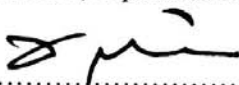
Wang-Cheol Song, Professor, Department of Computer Engineering



Seong Baeg Kim, Professor, Department of Computer Education



Namje Park, Associate Professor, Department of Computer Education



Thesis Supervisor

Yung-Cheol Byun, Professor, Department of Computer Engineering

December 2019

Department of Computer Engineering  
GRADUATE SCHOOL  
JEJU NATIONAL UNIVERSITY



# Acknowledgement

Foremost, I am grateful to Allah for giving me this opportunity, determination and strength throughout my life and ever more during the tenure of my research.

I dedicate this milestone to my beloved mother Bushra Shafqat and my father Shafqat Ali; I am forever indebted to them for giving me the opportunities and experiences that have made me who I am. I appreciate my close friend Sehrish Malik, for becoming my family abroad and always pushing me to do best in literally every department of my life. I wish to thank my sister Shifa Shafqat who has always selflessly encouraged me; and my brothers Ali Shafqat and Muhammad Umar Shafqat for always supporting me in my decisions. I would use this opportunity to thank my aunt Shameem Bano and my uncle Iftikhar Hussain Goraya for their continuous and unparalleled love, help and support.

I wish to express my sincere appreciation to my supervisor, Professor Yung-Cheol Byun, who convincingly encouraged me to be professional in achieving my goals and directed me to do the right thing even during my low times. Without his persistent help, the goal of completing this thesis wouldn't have been possible. I would like to acknowledge that he played a vital role in shaping my views towards maintaining a healthy balance between professional and personal life goals.

Finally, I wish to appreciate the support of all my friends and colleagues here; who are not just important in my life but also important to the successful realization of this thesis; also expressing my apology that I could not mention them personally one by one.

I greatly appreciate the assistance and exposure provided by Jeju National University and its Computer Engineering Department in fulfilling my dreams.

Dedicated to  
***My Sister, Shifa Shafqat*** ♥

# Table of Contents

Acknowledgement	iii
List of Figures	iv
List of Tables	vi
Abstract	1
Chapter 1: Introduction	3
Chapter 2: Related Work	9
2.1 Background Study .....	9
2.1.1. Topic Modeling	9
2.1.2. Recommendations System	23
2.1.3 Deep Learning and Topic Modeling	27
2.2 Limitations of Existing Solutions .....	29
Chapter 3: Proposed Model and Architecture	31
3.1 Conceptual Design .....	32
3.2 Topic Modeling based on LDA .....	36
3.2.1 Data Preprocessing	36
3.2.2 LDA Parameters and Configurations	37
3.3 Deep Learning Methods based on RNN-LSTM .....	40

3.4 Credibility Estimation .....	41
3.5 Overall Structure of the Proposed System .....	53
3.5.1 Input Data .....	55
3.5.2 Data Pre-processing .....	55
3.5.3 LDA and LSTM based Hybrid Model .....	56
3.5.4 Recommendations Module .....	56
3.6 Structural Details of the Hybrid Model.....	57
Chapter 4: Crowdfunding Project Recommendations: An Example Application	59
4.1 Experimental Data.....	64
4.2 Model (LSTM-LDA) Training.....	65
4.3 Project Integration.....	73
4.4 Objective Function Formalization for the Optimal Project Recommendations....	73
4.4.1 Optimization .....	78
4.5 Experimental Setup .....	79
4.5.1 Training .....	80
4.5.2 Testing .....	81
4.6 Example Scenario .....	82
Chapter 5: Experiments and Performance Analysis	87
5.1 Optimized Recommendations and Prediction Accuracies .....	87

5.2 Prediction Accuracy of Topic Classes .....	88
5.3 Prediction Accuracy of Topic Classes for Variable Number of Topics.....	88
5.4 Discussion Trends in Suspicious Campaigns.....	89
5.5 Analysis of Recommendation Results .....	91
Chapter 6: Conclusions	97
References	101



# List of Figures

Figure 1: Reviews based recommendation system development phases.....	7
Figure 2: An Example of Topic Modeling.....	11
Figure 3: Dirichlet multinomial modeling for short text .....	16
Figure 4: Types of recommendation systems .....	23
Figure 5: LDA based hashtag recommendation system .....	27
Figure 6: Textual and Non-textual Data based Reliable Product Recommendations.....	32
Figure 7: Conceptual diagram for hybrid LDA-LSTM mechanism .....	33
Figure 8: Layer-view for hybrid LDA-LSTM mechanism.....	35
Figure 9: Plate notation of LDA using crowdfunding comments.....	36
Figure 10: A detailed architectural view of LDA process in our system .....	38
Figure 11: Flow chart for PSO.....	52
Figure 12: Architecture of proposed hybrid approach.....	54
Figure 13. Input data explanation .....	55
Figure 14. Financial loss due to cyber scams in 2018 according to Internet Crime Report .....	60
Figure 15. The cyber scams victims in 2018 according to the Internet Crime Report. ...	60
Figure 16. An example of a scam campaign on Kickstarter .....	61
Figure 17. Comments on a Kickstarter Campaign.....	62
Figure 18. Data collection and selection process.....	63
Figure 19. Chronological order view for stored comments.....	64
Figure 20. Unrolling view for RNN's basic cell.....	66

Figure 21. Configuration Diagram for LDA.....	67
Figure 22. Topic classes categorization.....	71
Figure 23. Negative topic classes categorization.....	72
Figure 24. Example scenario input to model – Step 1 .....	82
Figure 25. Example scenario input to model – Step 2.....	83
Figure 26. Example scenario input to model – Step 3 .....	83
Figure 27. Example scenario input to model – Step 4.....	84
Figure 28. Example scenario input to model – Step 5.....	84
Figure 29. Example scenario input to model – Step 6.....	85
Figure 30. Example scenario input to model – Step 7.....	85
Figure 31. Example scenario input to model – Step 8.....	86
Figure 32. Example scenario input to model – Step 9.....	86
Figure 33. Prediction accuracy of topic classes vs. number of epochs.....	87
Figure 34. Topic classes and their respective prediction accuracy for variable no. of topics.....	88
Figure 35. Analysis of Topic classes with time in suspicious campaigns .....	90
Figure 36. Percentage accuracy of recommended results on ground truth data .....	91
Figure 37. Percentage accuracy of authenticity level of suspended projects.....	92
Figure 38. Percentage accuracy of authenticity level of canceled projects .....	92
Figure 39. Percentage accuracy of authenticity level of successfully-funded projects ...	93
Figure 40. Percentage accuracy of authenticity level of non-scam projects.....	94
Figure 41. Testing error against different learning rates .....	96

## List of Tables

Table 1: Topic modeling techniques with properties and limitations.....	13
Table 2: Topic models proposed in different fields.....	18
Table 3: Personalized recommendation systems, their properties and limitations .....	24
Table 4: Comparison of topic modeling and deep learning-based approaches with the proposed approach in Crowdfunding.....	29
Table 5: Parameters of LDA.....	39
Table 6: Neural Network Configurations .....	40
Table 7. Definitions of the parameters of authenticity .....	42
Table 8. All possible scenarios of project eligibility criteria.....	44
Table 9. Value ranges for each credibility parameters .....	50
Table 10. Algorithm for our proposed hybrid model.....	58
Table 11. Details of the implementation and experimental environment.....	65
Table 12: LSTM model configurations .....	67
Table 13: LSTM model configurations .....	68
Table 14. Identified topics classes after LDA analysis.....	69
Table 15. Classification of project’s credibility.....	75
Table 16. Parameters of PSO equation .....	78
Table 17. Experimental Setup.....	79
Table 18: Parameter settings for the optimization algorithm .....	95

Table 19: Classification accuracy of projects based on different batch sizes of comments

..... 98

# Abstract

The elevated growth rate of internet users and boom in the applications development, mainly e-business, has familiarized users to write their comments and reviews about the product they received. These reviews help customers in decision making. In this thesis, we aim to propose a recommender system that is primarily based on user comments or reviews. We aim to utilize the user-generated data to perform topic discovery and recommendations based on the topics discovered.

We take benefit of the language modeling practices and attempt to join them with neural networks (NN) to identify some latent discussion patterns in the feedbacks of users. Our goal is to design a probabilistic language modeling based neural network architecture, where RNN's special type Long-Short Term Memory (LSTM) model is used to predict discussion trends or topics. Probabilistic topic models utilize word co-occurrences across documents to identify topically related words. Due to their complementary nature, these models define different notions of word similarity, which, when combined, can produce better topical representations. We have trained our Latent Dirichlet Allocation (LDA) model with word embeddings to improve the topic quality. The hybrid of LDA-LSTM takes multiple inputs including words and topic embeddings, the temporal details, and probability distributions from LDA.

To capture the long contextual dependencies and limited vocabulary challenge, LSTM model takes both direct word embeddings and temporal topic distributions. Latent topic distributions are used to feed LSTM layers that are learnt based on the LDA model which is pre-trained. This proposed LDA-LSTM model, unlike previous studies, is capable of capturing both long range contextual and temporal dependencies. To enhance the recommendations results, we used Particle Swarm Optimization (PSO) as a baseline algorithm. For PSO, we have suggested an

objective function that takes into account the identified topic categories and user preferences. We have formulated an equation model to assess the credibility or authenticity of a recommendation. The proposed hybrid approach is experimented on crowdfunding campaigns and is used for reliable project recommendations. We collected data for scam and non-scam crowdfunding projects and applied our proposed approach to suggest secure and optimized recommendations to investors by using their comments.

For the proposed approach following metrics are considered for the performance analysis and evaluation of the proposed approach: i) prediction accuracy, ii) an optimal number of identified topics, and iii) the number of epochs. We compared our results with NN and NN-LDA based on these performance metrics. The strengths of both integrated models offer that the proposed model can play a significant part in an improved and effective understanding of user-generated data such as crowdfunding comments.

# Chapter 1: Introduction

With the upsurge of internet applications across the globe, people are familiar with the usage and advantages of these applications. One of the primary sources of this knowledge and expertise of people in using such applications is due to the availability of Social Network Services (SNS) and platforms such as Facebook, or twitter. One of the key contributions of these applications is the amount of data generated. This data helps in discovering different patterns and behaviors of users. People leave their comments or reviews more often these days about the products they have purchased or a place or restaurant they have visited. The user generated data is in such abundance that it becomes nearly impossible for a human being to extract the relevant knowledge out of it and draw some conclusions about the content.

Therefore, analysis of text data by using different Natural Language Processing (NLP) tools has become very popular. The basic language modeling techniques enable us to analyze the user generated data more effectively and in less time. Topic modeling being one of the elementary tasks of NLP is performed to discover hidden themes, keywords, and most discussed topics etc. In this work, we have used topic modeling for discovery of topics. The potential applications of topic modeling are widespread in different fields such as in bioinformatics [1], image or document analysis [2], content recommendations [3-6] and measuring service quality [7] etc.

We aim to use topic modeling approach for user generated data to discover latent topics and latest trends in discussions, which is very important for any system. Many service providers are using such techniques to improve their content quality according to their customer's requirements. Though, in crowdfunding, a platform to raise funds for a novel concept or idea for some rewards or a complete product in return, the comments are frequently overlooked. Therefore, we have used crowdfunding for experiments. First, we aim to take advantage of topic modeling approaches to

discover different trends and topics by targeting crowdfunding comments. Besides that, the comments content can evidently become a significant area in serving crowdfunding sites to fight against the major risks of deceitful events.

Second, we aim to implement a Recommendation System (RS) that uses the discovered topics, learns about user's preferences and then recommend a product to the user. Here, a product can be anything such as a book, a tourist place, or a crowdfunding project. As we are using crowdfunding for experiments, the product will be a crowdfunding project in our scenario. There is no denial in the fact that RSs are extensively used in various fields primarily in e-commerce to offer tailored recommendation services for users. RSs are getting advanced and intelligent at a very rapid pace. These systems help in overcoming the challenge of information and data overload by offering customized suggestions to users based on the data generated by the users in form of likes, ratings, reviews etc. The trendiest and commonly used RSs are based on Collaborative Filtering (CF) [8]. However, there are also some limitations to CF models such as data sparsity. Data sparsity refers to the case when the available data is short, the model performs poorly. The issue of data sparsity can be handled by using more data such as reviews, comments, blogs related to one specific item. This way, data can be increased and more knowledge can be withdrawn from it. Also, the user comments or reviews reflect a lot about a user's preferences and emotions.

Our objective is to propose a language modeling-based neural network structural design. Here, Long-Short Term Memory (LSTM), a Recurrent Neural Network (RNN) model is used to acquire and classify the comments into one of the topics cases learnt by the topic modeling mechanism. We are using Latent Dirichlet Allocation (LDA) as a topic modeling approach. LSTM then trains the network based on the results of LDA to predict discussion trends such as either scam or non-scam. The latent topic probability distributions and word probability distributions learnt from the pre-trained LDA model are used as a feed to LSTM layers. After the classification and prediction task, we use optimization algorithm to optimize our results that is Particle Swarm



Optimization (PSO). This adds an extra layer to the system that estimates the credibility of a product before recommending it to the user. The credibility estimation process is also based on user preferences. In case of crowdfunding, it helps the investors to find reliable projects to pledge in i.e., a project with maximum chances of delivery. We have used different parameters and metrics for performance measurements such as prediction accuracy, an optimal number of identified topics and the number of epochs.

Previously proposed approaches are primarily using topic modeling to understand user comments or reviews. Another efficient way to understand users' comments is to integrate the topic modeling approach such as LDA with deep learning methods such as CNN or RNNs. Though CNN learns word contexts and orders, it fails to detain the long-range semantic dependencies because of the small window size. With RNN, we can better understand the comments. As RNN takes a sentence as series of words, plus the information related to the previous state, it increases its capability to retain the complete sentence structure. Though the performance will improve with RNN, it might face some limitations when the sentence becomes too long [9-10].

Therefore, we propose a hybrid approach based on deep learning and topic modeling to extract more global contextual information for deeper understanding of user-generated data. The motivation behind this proposed approach is to overcome the limitations of the previously proposed mechanisms. Deep learning methods can learn deeper contextual dependencies while topic modeling can give word co-occurrence relation to make a supplement for information loss.

We use LSTM network for the deep learning as it is a special RNN type that shows enhanced performance results than vanilla RNN as it overcomes the gradient vanishing and long-term dependence problems. We use LDA to perform topic modeling. It results into different clusters of words, each representing a theme or topic. Along with other outputs from LDA, the two parts are integrated into a single system. The final model is named as a hybrid LDA-LSTM model.

Furthermore, as the topic modeling part and deep learning part are joined in our model, the topic clustering results will be influenced by the deep learning information.

The hybrid model is then layered with an optimization module that estimates the credibility of an item to be recommended. This module performs optimizations based on objective functions provided by Particle Swarm Optimization (PSO) algorithm.

In summary, there are many developed applications for recommendation systems in different fields. Our proposed approach is a novel approach to recommend a credible item as far as we know. Moreover, none of the works have focused on crowdfunding comments to find discussion trends and their impact towards project credibility. Hence, in crowdfunding, this approach can be used to recommend safe or secure projects to investors.

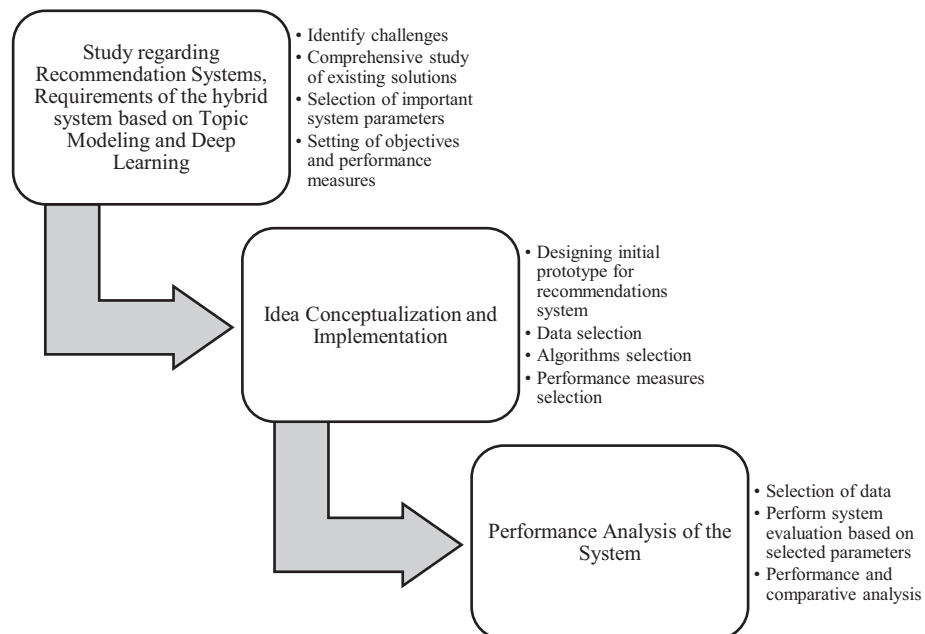
The objective of this study is to overcome the limitations of topic models and deep learning and get the most out of both approaches. The main objectives include:

- 1) Finding ways to preserve the contextual dependencies as traditional topic models are based on bag-of words approach, so there is high probability to miss the contextual and temporal dependencies.
- 2) Recommendation tools are in use since a long time now, finding credibility of the recommended product or location is a potential target of this research.

A brief summary of the contributions of this thesis can be listed in the following manner: (1) we propose a generic and hybrid approach towards secure recommendations of crowdfunding projects. A dynamic word embeddings-based procedure that can model user and word embeddings in a joint and dynamic manner. It can model these embeddings in the similar contextual space for a stream of documents. By this, we are able to measure the semantic similarity in a very effective way among users and words. (2) The proposed algorithm can infer both the dynamic embeddings for words and for the documents. Our optimization module works based on an objective function.

We propose a credibility measurement approach for secure recommendation (4) We validate our proposed model and its effectiveness, the optimization algorithm, and our objective function, on crowdfunding projects and tourism blogs. We compared our proposed model with baseline algorithms as NN-LDA, NN, and SVM-LDA etc. The results prove that our proposed approach is significantly better than the other state-of-the-art techniques.

Research methodology adopted for this study have three main phases as shown in Figure 1 below.



**Figure 1: Reviews based recommendation system development phases.**

Phase 1: Study regarding Recommendation Systems, Requirements of the hybrid system based on Topic Modeling and Deep Learning

- Identify challenges
- Comprehensive study of existing solutions
- Selection of important system parameters
- Setting of objectives and performance measures

## Phase 2: Idea Conceptualization and Implementation

- Designing initial prototype for recommendations system
- Data selection
- Algorithms selection
- Performance measures selection

## Phase 3: Performance Analysis of the System

- Selection of data
- Perform system evaluation based on selected parameters
- Performance and comparative analysis

The rest of the thesis is structured and organized in the following way: Chapter 2 presents a brief overview on related studies. We have discussed related works of all the fundamental components and algorithms in this chapter and all algorithmic approaches used for the experiments. Chapter 3 is related to the proposed system design and conceptualization. Chapter 4 discusses the simulation results where we have discussed in detail crowdfunding as a potential example of our proposed approach. Chapter 5 presents comprehensive experimental analysis. Results are categorized into four subsections (a) training results of learning algorithms (b) analysis of prediction results (c) analysis of optimization results and (d) analysis of the recommendation results. Chapter 6 concludes this thesis along with the future directions.

# Chapter 2: Related Work

In this section, we have presented the literature review of the related works of recommendation systems and topics discovery. Topics discovery is one of the key tasks of NLP. As the amount of data available increases, it becomes more time consuming and tiring to get the gist out of it in a short period of time. Therefore, topic modeling helps in this regard. It is also very useful for an effective recommendations system.

We have divided the related works into three sub-sections. In section 2.1 we present the background study that includes works on topic modeling in section 2.1.1, recommendation systems in section 2.1.2 and deep learning and topic modeling-based approaches in section 2.1.3. In section 2.2 we present limitations of the existing solutions. This chapter as whole covers all the related and existing studies comprehensively primarily related to topic modeling and LDA, deep learning and recommendations, and optimization. For this research work, many papers related to topic modeling are investigated.

## 2.1 Background Study

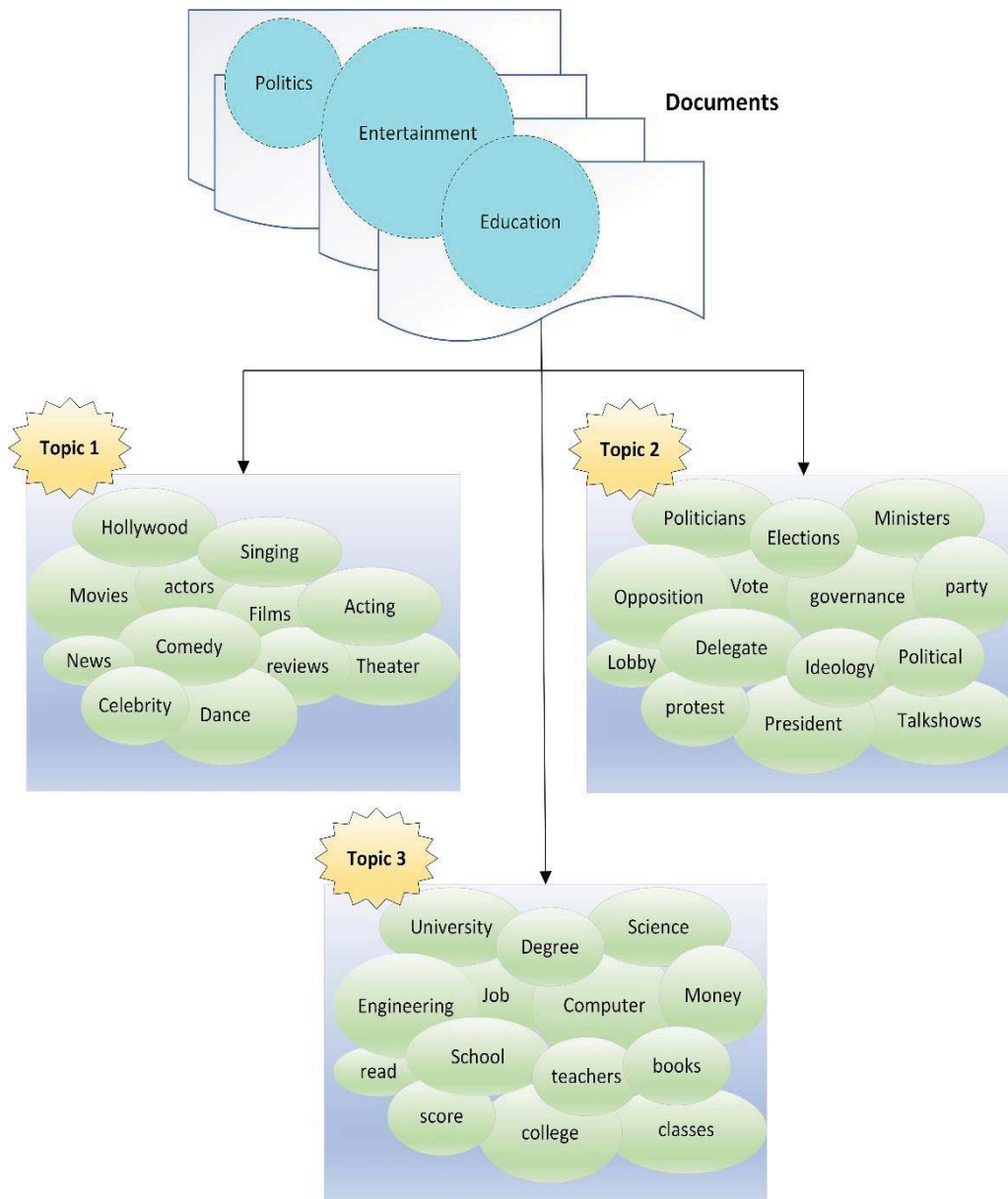
This section discusses the required background knowledge. It is vital for the reader to recognize the rest of the thesis. This segment will briefly discuss the algorithms and ideas used for the experiments. In section 2.2.1 topic modeling is covered which includes topic modeling with LDA and in the proceeding sections topic modeling for the recommendation systems is discussed.

### 2.1.1. Topic Modeling

In this era of internet and digitalization, an enormous amount of textual data is being generated at an extremely high rate. The significance of studying and analyzing textual statistics has emerged

with the rate at which data science is got fueled with big data. The applications of text data analysis are widespread starting from customers reviews analysis to extracting and finding hidden meaning of a large dataset. A novel approach is proposed by Blei to recognize the topics which ultimately led to sentiments classification, documents classification, and unlocked relatively many assessment prospects for textual data [10].

Topic modelling is a very simple concept which focuses on to identify the themes inside a corpus. It basically is a probabilistic method to discover the topics which is very time consuming for bare human eye. The concept ‘topic modeling’ is aptly named; it is a technique to automatically generate topics from a corpus that offers insight on the latent patterns and themes of the dataset. Unlike the regular expression-based methods or rule-based techniques on which some general rules are generated to discover the significance of the words in a document, topic modeling is more of a probability and statistics-based technique. In order to understand it in a simple manner, let’s take an example.



**Figure 2: An Example of Topic Modeling**

Let's suppose that we have a large amount of textual data available which covers different subjects like politics, entertainment, and education as shown in Figure 2. A topic modeling approach will generate clusters of words based on their probabilistic values to be in a specific topic.

Therefore, after the topic modeling approach being applied to the dataset, following key topics will be generated- Topic 1: Hollywood, movies, reviews, drama, singing, acting, actor, films, celebrity, news, comedy, theater etc. Topic 2: Politicians, ministers, party, opposition, protest, vote, elections, lobby, delegate, ideology etc. Topic 3: Computer, science, school, university, engineering, college, books, read, score, classes, job, teachers, degree etc.

This a very basic example of how topic modeling technique extracts unseen semantic structures of a data. These structures and themes are identified without going through all the data. This characteristic of topic modeling helps in different classification tasks such as sentiment classification, document classification. It also helps in predictions and recommendations as it discovers hidden patterns. The impacts of topic modeling are not limited to classification and prediction problems, it is applicable in different fields of data science and effective in simplifying different NLP tasks. This thesis has focused on topic modeling, deep learning as well as their applications in recommendation systems.

Topic models are of key importance for the illustration of discrete data, and are used in different research fields such as medical sciences [11], software engineering [12], geography [13], and political sciences [14] etc.

There are many topic modeling techniques, each have its own strengths and limitations. The most frequently used approaches include latent semantic analysis (LSA) [15], probabilistic latent semantic analysis (PLSA) [16], latent Dirichlet allocation (LDA) [17] and correlated topic model (CTM) [18].

The Table 1 below introduces different topic modeling methods along with their properties and limitations.



Table 1: Topic modeling techniques with properties and limitations

Topic Modeling Method	Properties	Limitations
Latent Semantic Analysis (LSA)	<ul style="list-style-type: none"> <li>- Generates a vector-based representation for texts to make semantic content</li> <li>- Statistical background is not robust</li> <li>- If there are any similar words, LSA can get from the topic</li> </ul>	<ul style="list-style-type: none"> <li>- Number of topics are difficult to be determined</li> <li>- Loading values and their interpretation with the meaning of probability is difficult.</li> </ul>
Probabilistic Latent Semantic Analysis (PLSA)	<ul style="list-style-type: none"> <li>- All words can be generated from a single topic.</li> <li>- In a given document different words can be generated from multiple topics</li> <li>- It handles polysemy</li> </ul>	<ul style="list-style-type: none"> <li>- It fails to perform probabilistic modeling at document level</li> </ul>
Latent Dirichlet Allocation (LDA)	<ul style="list-style-type: none"> <li>- Text preprocessing has to be done separately such as removal of stop words</li> <li>- It is used as a statistical and probabilistic model.</li> <li>- Number of topics can be selected.</li> </ul>	<ul style="list-style-type: none"> <li>- It is hard to generate relationship between topics by using LDA</li> </ul>
Correlated Topic Model (CTM)	<ul style="list-style-type: none"> <li>- It uses logistic normal distribution to draw relationships among topics.</li> <li>- Same words are allowed to occur in different topics at the very same time</li> </ul>	<ul style="list-style-type: none"> <li>- Calculations are huge.</li> <li>- Topics can have many general words which might impact the topic quality.</li> </ul>

LSA is an NLP approach. The primary focus of LSA is to generate different representations of texts based on vectors to create semantic content [19-20]. These vector representations are created to choose related words by computing the similarity among text data. LSA has many applications such as keyword matching, word quality assessment, power collaborative learning, guidance in career choices, making optimal teams etc. [21]. Also, it uses singular value decomposition (SVD) to reorder the corpus. Other applications of LSA can include reduction of dimensions [22], and identification of research trends [23].

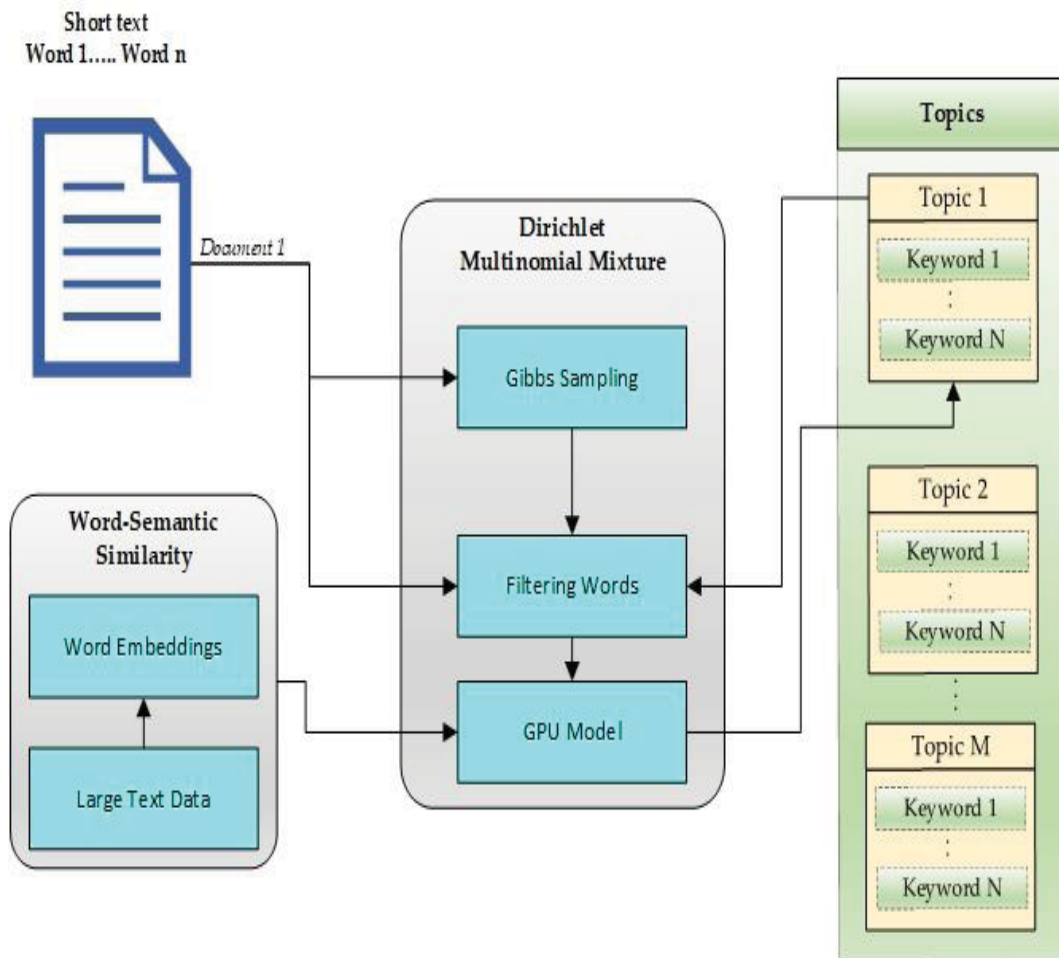
PLSA is a technique that has been introduced after LSA method to fix the limitations LSA. It was first presented by Jan Puzicha and Thomas Hofmann in year 1999 [24]. This approach can automate the process of document indexing and uses a generative model as an advancement to LSA in a probabilistic manner. The primary goal of PSLA is to discover and discriminate between various situations in which words are used without using any dictionary. It has many implications which mainly includes the differentiation of the words with several meanings and clustering words that share similar context [25]. In [26], PSLA is referred as an aspect model which is based on a latent variable that is responsible to link each observation with unseen class variables. Besides introducing advancements in LSA, PSLA has many other applications which include recommender systems and computer vision [27]. Other applications of PSLA include image retrieval [28] and automatic question recommendations [29].

LDA model was introduced to overcome the limitations of LSA and PSLA in capturing the exchangeability of documents words. The amount of data available online in form of web data, blogs, news, articles, and literature has introduced multiple challenges to data mining and computation tasks. With the increase of such data, the need for the tools to automate the data visualization, analysis, and summarization of web documents has increased. LDA being an unsupervised approach for topic modeling has recently become very popular mainly for topic discovery in large corpus. In [30], LDA is used for text mining that is based on Bayesian topic

models. LDA is also a generative and probabilistic model that attempts to imitate the writing task. Therefore, it attempts to produce a document if a topic is given. There is a variety of LDA based algorithms used in different domains including: author-topic analysis [31], LDA based bioinformatics [30], temporal text mining [32], supervised topic models, and latent co-clustering etc. In simple words, the key idea of LDA is, each document is represented as a mixture of topics, each topic represents a discrete probability distribution reflecting the likelihood of each word to be occurred in a specific topic. Therefore, a document is represented as probability distributions of words in each topic. Certainly, LDA has many applications such as role discovery [33], emotion topic [34], automatic grading of essays [35], and email filtering [36] etc.

Correlated Topic Model (CTM) is a statistical model used in machine learning and NLP. It is used to identify the topics revealed in a group of documents. CTM depends on LDA and it uses logistic normal distribution.

Biterm Topic Modeling (BTM), is a topic modeling approach over short texts. This topic modeling methods is becoming a significant job because of the pervasiveness of the short texts available on the internet. With the emergence of social media services and applications, today short texts are of great significance. As abundant of short text data is available, it is very critical to effectively discover topics out of it. The short texts understanding required by many applications becomes critical and important process mainly deducting and discovering discriminative and comprehensible latent topics from it. Also, the traditional topic modeling approaches greatly depend on the co-occurrences of the words to deduct topics from a corpus. Therefore, in [37] BTM is used specifically for this purpose. BTM can effectively detect topics inside the short texts through exhibiting patterns of co-occurrences of the words cooccurrences in dataset. It analyzes the generated biterns and collects the hidden topics as a set of text-documents and distribution.



**Figure 3: Dirichlet multinomial modeling for short text**

The results have shown that the BTM creates comprehensible topics in short texts as well as discriminative topic depictions. By learning the global topic distributions, BTM prevents the issue of data sparsity [38]. Figure 3 provides an overview of a model for short text sampling for topics discovery that encourages words that are semantically related under a given topic. In this scenario, words that are semantically close are connected together, regardless of their level of co-occurrence in the short texts. To control the inference of topics, a filtering module is introduced. In the Figure, the model is based on a generalized Polya's urn (GPU) approach which uses Dirichlet Multinomial Mixture (DMM). These models are both fast and flexible due to the fact that word embeddings can be pre-learned from large text datasets. The proposed GPU-DMM model was experimented

on two different languages dataset, and the topic representations learned produce the best accuracy for classification of text.

Topic modeling approaches are applied in various text mining and NLP tasks. In computer science topic models are applied on texts such as news, comments, tweets, emails or books etc. Topic models include generative probabilistic models for text [39] and statistical language models [40]. Some other types include bigram language model [41] and Hierarchical Dirichlet Language Model (HDML) [42]. There are some extended versions of these approaches such as HDML predictive distribution [43], and LDA [17] etc. Some other relevant studies have also focused on problems related to fields such as aspect and opinion mining [44-47], image classification [48-50], source code analysis [51-55], emotion classification [56-59], event detection [60-63], and recommendation systems [64-68].

#### ***2.1.1.1 Topics Discovery with LDA***

There are many different approaches for obtaining topics from a set of text data. Topics can be generated from Term-Frequency and Inverse-Document frequency (TF-IDF) model, non-negative matrix factorization etc. However, Latent Dirichlet Allocation (LDA) is the most popular technique applied on topic modelling. This approach has never been explored on the Bangla language. LDA is basically a probabilistic algorithm that generates topics from a Bag of Words model. Before the LDA is discussed, a general overview of Topic and Term is given as follows: A topic is a collection of words with different probabilities of occurrence in a document talking about that topic. If there are multiple documents, a topic would consist all the words initially from that collection. After the model is trained the topics will consist of words that are highly relevant. A term is a word in a topic. Each term belongs to a topic. The LDA algorithm generates a topic-term matrix. LDA is a matrix factorization technique [69]. At first, it generates a document-term

matrix. In the matrix, assume that there are two documents d1 and d2 and two words t1 and t2. So, the example matrix is as follows:

In this matrix, the rows represent the documents and the columns represent the terms t1 and t2. 0 and 1 shows if the word belongs to the document or not. Then the LDA generates two more low level matrices representing document-topic and topic-term matrix which will be discussed in detail in chapter 3. Afterwards, a list of all the unique words are made from all the documents. The algorithm goes through each word and adjusts the topic-term matrix with a new assignment. A new topic “K” is assigned to the word “W” with a probability P [17]. The procedure of calculating the probability will be discussed in the related section. The algorithm keeps iterating until a situation is reached when the probabilities do not change considerably which means that the probabilities will have a very small fractional change with further iterations. At this point the algorithm converges and is expected to have the topics extracted.

In Table2, we provided a classification of topic modeling methods based on LDA, from some of the impressive works.

**Table 2: Topic models proposed in different fields**

Year	Model	Description
2009	MedLDA	- maximum entropy discrimination LDA [70]
	Relational Topic Models	- Predict links between documents [71]
	LACT	- An LDA based approach that categorizes software-based solutions automatically. This approach is primarily used in open source repositories for

		the categorization of software systems [72]
	Labeled LDA	- This model is also based on LDA where the correspondence among latent topics and the tags of users is one-to-one [73]
	HDP-LDA	- Nonparametric Bayesian approach to clustering grouped data [74]
	LDA-SOM	- This approach generates clusters of documents that is based on the topics. It also makes clusters in a subjective 2D format [75]
2010	Topic Aspect Model	- Bayesian mixture model which jointly discovers topics and aspects [76]
	Geo-Folk	- Location based Images [77]
	TWC LDA	- Constrains different topics to be weak-correlated [78]
	Dependency Sentiment LDA	- This approach makes an assumption that the topics within a document can form a

		Markov chain. Secondly, the topics of the sentence plus the topic of its previous sentence can have an impact on the sentiments of a given sentence [79]
2011	Sentence LDA	- Topic Model for Sentences [80]
	Twitter LDA	- Event Identification from Twitter Textual Data [81]
	SHDT	- Symptom-Herb-Diagnosis topic [82]
	LogicLDA	- Topic modeling with First-Order Logic (FOL) domain knowledge [83]
	MMLDA	- Statistical topic model for the task of image and video annotation [84]
	Bio-LDA	- Extracted biological terminology to automatically identify latent topics [85]
2012	Mr. LDA	- Uses variational inference, which easily fits into a distributed environment [86]



	LDA-G	- Identify the anomalies in network traffic [87]
	sshLDA	- Obtain hierarchical topics [88]
	ET-LDA	- This model uses a joint statistical modeling of topical influences approach that takes events and their related tweets to align an event and its tweets [89]
2014	emotionLDA	- For recognizing emotion [90]
	Biterm Topic Modeling	- This approach is used for short texts that is based on a Biterm topic model [37]
2015	LightLDA	- This approach is introduced for problems where big topic models or deep NNs have to be implemented. It focuses on reducing the computation complexity and massive requirements [91]
	Latent Feature LDA	- This approach is to improve the performance of traditional topic models by using word

		representations as a latent feature [92]
	MRF-LDA	- This approach is a topic modeling approach for short texts that also uses word embeddings [93]
2016	Hashtag LDA	- This approach is a hashtag-based approach for discovering sub-events in Twitter and it uses Mutually-Generative LDA model.
	Embedded-LDA	- Combing LDA and Word Embeddings for topic modeling [94]
2017	iDoctor	- This approach is used for medical recommendations that are personalized and as well as professionalized built on hybrid matrix factorization [95]

## 2.1.2. Recommendations System

Recommendation System is an intellectual system that makes suggestion about items to users that might interest them. Some of the practical example applications of RSs include movie, book, tourist spot recommendations, etc. It's a point of amusement to discover that how "People you may know" feature on Facebook or LinkedIn.

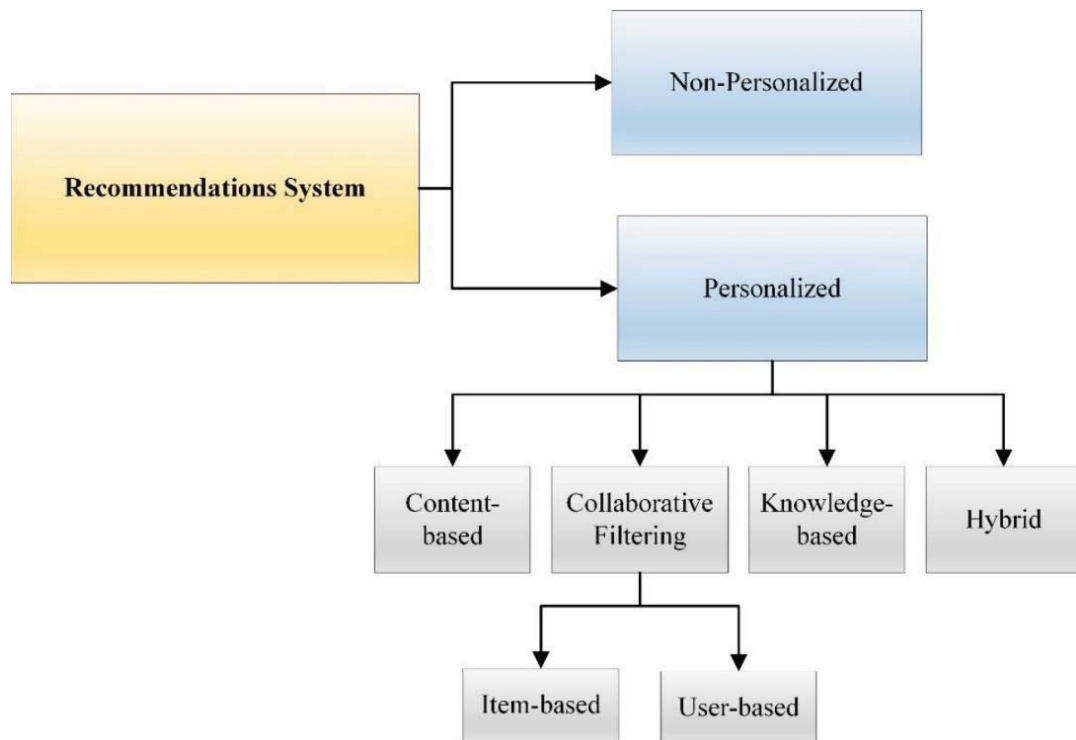


Figure 4: Types of recommendation systems

This characteristic of RS systems presents people you might have known, who are comparable in terms of some common interests, or common friends or may be related to your friends of friends, or may share the same location or are probably nearby, might have same skills or talents, and so on.

Figure 4 shows the types of RSs and what techniques they use, therefore any RS usually produces a number of recommendations based on one of the techniques mentioned.

In a personalized RS, users are suggested items based on their past behaviors and also their social networks based interpersonal relationships. It primarily focuses on following perspectives: 1) To discover whom you can trust referred as Interpersonal impact, 2) Finding users of similar interests referred as interest circle derivation, 3) Individual interests of user which discovers what sort of items a person has passion for. There are primarily 4 fundamental categories of personalized recommendation systems based on the recommendation approach being used as shown in Table 3.

**Table 3: Personalized recommendation systems, their properties and limitations**

<b>Personalized Recommendation systems</b>	<b>Properties</b>	<b>Limitations</b>
Content-based Filtering	<ul style="list-style-type: none"> <li>- It does not require data of other users.</li> <li>- There are no data sparsity issues</li> <li>- It does not face cold start problem</li> </ul>	<ul style="list-style-type: none"> <li>- In order to define the features of an item, content analysis is important.</li> <li>- Product's excellence cannot be measured.</li> </ul>
Collaborative Filtering (CF)	<ul style="list-style-type: none"> <li>- A product's excellence can be measured by using user's ratings</li> </ul>	<ul style="list-style-type: none"> <li>- Cold start problem exists for new products and different users.</li> <li>- There is a tradeoff between stability and plasticity problem</li> </ul>
Knowledge-based	<ul style="list-style-type: none"> <li>- User's preferences are taken into account</li> <li>- There is constraint based and case-based models</li> </ul>	<ul style="list-style-type: none"> <li>- It greatly depends on what kind of data about the item is used into the model.</li> <li>- In case of less information, it might not perform well.</li> </ul>

Hybrid	<ul style="list-style-type: none"> <li>- Any models can be combined to find personalized business solutions.</li> <li>- Combines the merits of any two approaches to overcome challenges if faced any.</li> </ul>	<ul style="list-style-type: none"> <li>- Discovering the way to combine methods is challenging</li> <li>- Its challenging to discover that for good predictions what number of systems should be combined and how</li> </ul>
--------	---	--

In [96], user-based CF algorithm is implemented on Hadoop to solve the problem of scalability in CF methods. This proposed approach performs better for discovering the interests for similar products and also provides personalized recommendations. One disadvantage of this approach is that it fails to consider the interests of similar users. In [97], a personalized travel recommendation system is proposed and presents a promising application by using the publicly available photos shared by people. In addition to that, customized travel recommendations are proposed by taking into account the specific user profiles and considering the attributes such as travel group types (e.g., friends, single, couple or family). The biggest advantage of considering community contribution as an attribute for effective recommendations. Also, from shared photos lots of other attributes can be derived. On the other hand, its computational complexity becomes a disadvantage along with the privacy issues raised while processing the photos contributed by the community. An item-to-item CF approach is proposed in [98] to extract useful and interesting videos out of largescale of videos. This methodology is implemented in a MapReduce framework. It enables with better recommendations by using interests of similar users for a specific item. It also does not consider the similar interests and it is complex to be implemented. In another study [99], a personalized recommendation system is proposed. It is also based on user-based CF algorithm. It is implemented on Hadoop to generate more effective and scalable results. For evaluations, author has used Jaccard coefficient and Cosine similarity measures. It performed better than the

conventional methods. The primary advantages of this approach are its scalability and efficiency. There are a couple of its advantages of this approach as the Jaccard coefficient method used is not highly accurate, also the user's reviews are not separated into positive and negative ones.

A novel clustering method is proposed in [100] that uses latent class regression model as a baseline model which considers both the general ratings and textual reviews. In [101], a system that considers the location of a user as an attribute of a recommendation system is proposed. This proposed system performs better for location-based services and it also minimizes the transmission cost. On the other hand, it is not best suited for the large datasets and where location is not of any concern to the user. A recommendation method is suggested in [102] which investigates the difference between user feedbacks to discover the preferences of a customer. It considers user ratings and also focuses on the sparsity issue of the data. In addition to that, online feedbacks of restaurant clients were also investigated to generate a RS for restaurants and prove the efficiency of the proposed method. In [103], a CF method is being suggested that uses ratings of different items and feedbacks on different social networks such as twitter. It generates suggestions for different items.

A bulk of scientific articles are accessible by the researchers. It results in greater difficulty of finding papers that are relevant or related to their respective field of study. There are novel ways proposed to solve this issue by recently shaped online societies of researchers, i.e., sharing citations. In [104], an algorithm is developed that is for online communities and it recommends the scientific articles to the users of these communities. The advantages of the state-of-the-art CF method and probabilistic topic modeling are combined here. It offers an explainable latent configuration for both users and the items. It can also generate recommendations for both present and recently published articles. For example, in recommendation system, in [68] proposed a hashtag recommendation scheme based LDA model that can discover hidden topics in microblogs. The complete conceptual design is also shown in Figure 5.

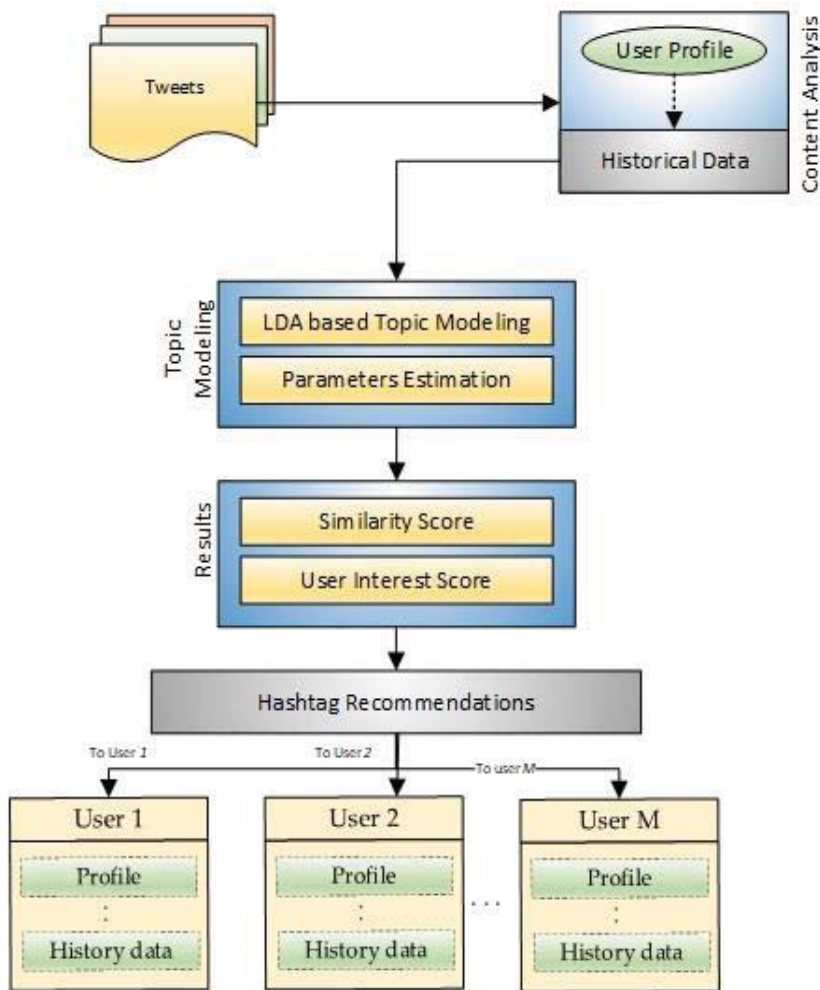


Figure 5: LDA based hashtag recommendation system

### 2.1.3 Deep Learning and Topic Modeling

The terms used to define RNN are,  $\Sigma$  which indicates inputs,  $S$  representing states and  $\delta$  represents the transition function of NN. In RNN, a document is considered as a sequence and an LSTM gets trained with previous words to forecast the succeeding word by maximizing  $p(w_t|w_{t-1}, w_{t-2}, \dots, w_0; \text{model})$ . For updating states in LSTM, the input indicated by  $\Sigma$  are converted into vectors. Activation function such as Softmax is led by a vector of the size of dictionary which involves the inclusion of  $st$  for the result or output. Problem arises when the dictionary size increases enormously.

Oh et al. proposed a model for user profiles using deep neural network that was altered from DBN to obtain extreme accuracy. Term Frequency (TF), Title (T), Inverted Term Frequency (ITF), Cumulated Preference Weight (CP) and First Sentence (FP) are the features for the input layers. Values of input features like TF and ITF are greater than 0, Boolean values for T and FP and value of CP is between 0 and 1. The author used 3-layer perceptron calculation, to achieve the user's profile. A CNN, devised by Krizhevsky et al. is referred as deep CNN [105] that learned 1000 semantic concepts for training based on ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 dataset [106]. This CNN had a complex structure with 5 convolutional and 3 fully connected layers and the results were represented by using feature representation of the sole images. Likewise, for the knowledge of visual features Lin et al. utilized deep CNN. Deep CNN proposed by [105] is not good for clothing domain. Therefore, fully connected layers have been included between 7th and 8th layer to fill the gap between semantics and mid-level features. This helps in learning binary functions and representation of clothing images. To find the neighborhood relationship of the user profile Cui et al. developed a DBN model. Initialization and tuning are the two stages for training DBN that has multilayer RBM. Markov Chain Monte Carlo (MCMC) process in the DBN model is developed to build the transfer function by forming the maximum likelihood law technique that uses Gibbs sampling. The proposed architecture by Lei et al. was named as comparative deep learning architecture to obtain the comparative distance between a user and two images. One sub-network for user and two sub-networks for positive and negative images are used in the CDL model. In [107], author built a CNN model for the classification of the music genre. This model comprises of 2 convolutional layers, 1 fully connected layer and 2 max-pooling layers. Also, there are 10 softmax units with a logistic regression layer to classify the music genre.



There is another deep RNN (DRNN) method for e-commerce system for recommendations in real-time and it builds user's history. This DRNN is influenced from [108], but has many differences. The DRNN tracks the user's browsing patterns.

To find the desired result, users often visit many pages. The DRNN model aims to build a real-time recommendation to reduce the number of web pages for efficient web search. Sparse autoencoder was proposed by Zuo et al. to process tag information.

## 2.2 Limitations of Existing Solutions

Most of the studies in literature are focused on simple embeddings and haven't considered to use words plus topic embeddings for the LSTM training. We have incorporated these embeddings to make more contextual aware recommendations. Also, the crowdfunding case study is an addition to the research in crowdfunding. As far as our literature survey is concerned, there are no such studies that has used similar approaches and targeted crowdfunding to detect suspicious or fraudulent activities.

In Table 4, we have presented different studies done on crowdfunding, and focused on what platforms they have used, if they used language analysis or not, if they have performed topic modeling using LDA or not, if the hybrid of LDA and LSTM is used or not, if any optimization is performed or not, if the study has used crowdfunding comments data or not, and is this study detects fraudulent activities or not. Hence, Yes shows the existence of the particular feature and No shows the absence of that particular feature.

**Table 4: Comparison of topic modeling and deep learning-based approaches with the proposed approach in Crowdfunding**

<b>Related Works</b>	<b>Platform</b>	<b>Language Analysis</b>	<b>Topic Modeling using LDA</b>	<b>Hybrid (LDA-LSTM)</b>	<b>Optimization</b>	<b>Comments</b>	<b>Fraud detection</b>
[109]	Kickstarter	Yes	No	No	No	No	Yes
[110]	Tourism Crowdfunding	No	No	No	No	No	No
[111]	Kickstarter	No	No	No	No	No	No
[112]	Kickstarter	Yes	No	No	No	No	No
[113]	Kickstarter	Yes	Yes	No	No	No	No
[114]	Kickstarter	Yes	Yes	No	No	No	No
[115]	Kickstarter	Yes	Yes	No	No	No	No
[116]	Kickstarter	Yes	No	No	No	No	Yes
[117]	Kickstarter	Yes	No	No	No	Yes	Yes
[118]	Kickstarter	No	No	No	No	Yes	No
[119]	Medical Crowdfunding	No	No	No	No	No	Yes
Proposed approach	Kickstarter	Yes	Yes	Yes	Yes	Yes	Yes

# Chapter 3: Proposed Model and Architecture

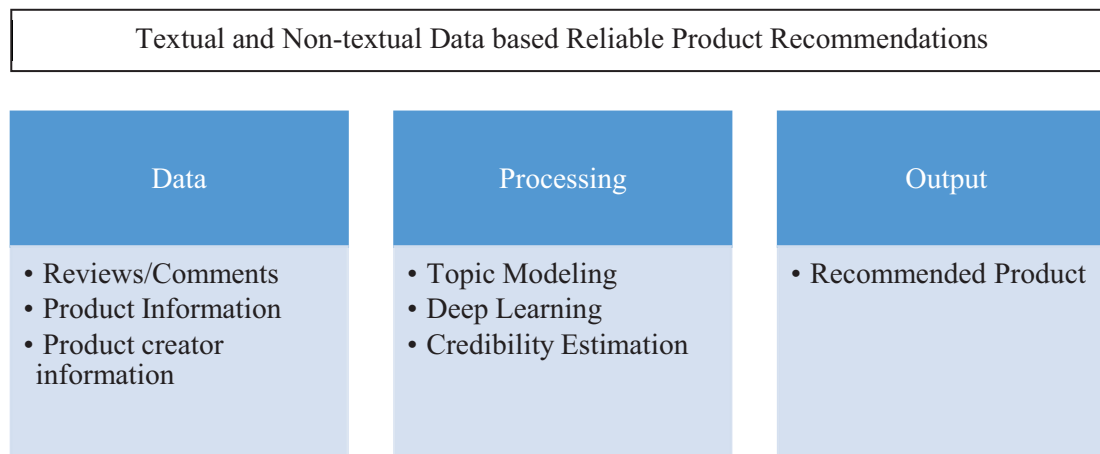
In this chapter, comprehensive description of the proposed hybrid approach is presented. A topic modeling scheme combined with deep neural networks is referred as a hybrid approach yields topics for semantically similar documents. We propose the topic modeling-based LSTM model, which blends LSTM and topic modeling for topics discovery and recommendations. The proposed model is able to extract more comprehensive context information and a broader understanding of user generated data. The context information is reserved through deep learning methods, whereas the information loss is controlled through topic modeling which can support word co-occurrence relations.

In section 3.1, the theoretical framework of our system is presented. In section 3.2 the complete hybrid approach is presented which covers input data details, its preprocessing, and recommendation module. In section 3.3, the hybrid model structure is presented, in section 3.4, the model preliminaries are presented by covering LSTM and LDA details, next section focuses on the key modules of the system.

The model utilizes both non textual and reviews data. For the non-textual part, we crawl the static information provided and calculate the statistics. For the textual data part, we use LDA to extract topic latent vectors and adopt an LSTM architecture to generate document latent vectors.

The basic flow of the system is illustrated in the Figure 6. The first step is data collection that can be any reviews or comments, any product or project related information and the information related to the content creator (either the product or project creator). The next step is

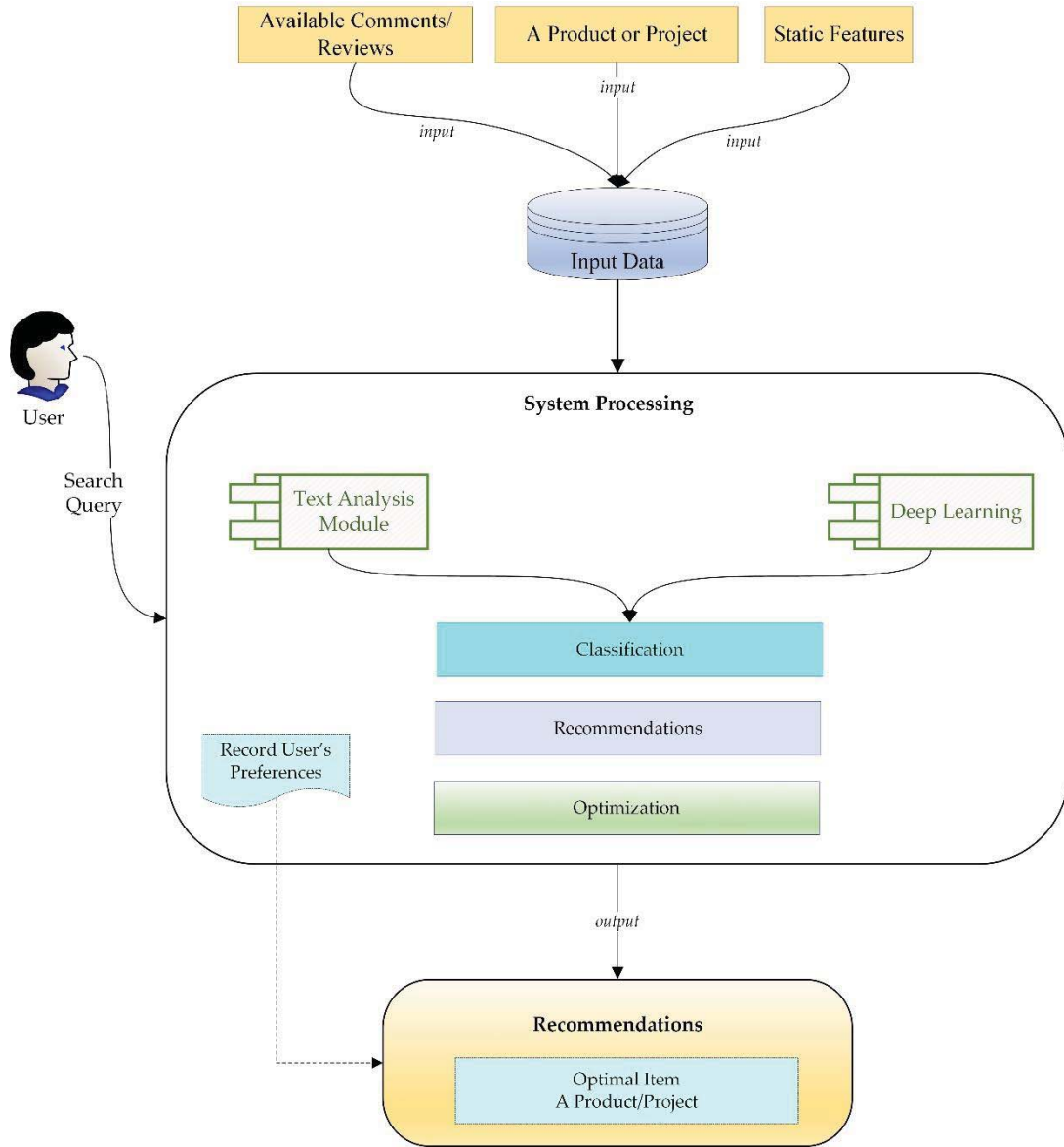
to process the raw data. This step comprises of preprocessing of the raw data that prepares the data to be used to feed the topic modeling module and deep learning module. It then performs the credibility estimation and finally the product is recommended based on the credibility and user preferences.



**Figure 6: Textual and Non-textual Data based Reliable Product Recommendations**

### 3.1 Conceptual Design

The proposed system is for recommendations that does the language analysis, learns the hidden patterns and acquires the user preferences. The proposed system can work for any kind of product that has some characteristics along with user ratings, reviews and other static features such as published date and number of reviews etc. Figure 7 presents the basic conceptual flow of the proposed system.



**Figure 7: Conceptual diagram for hybrid LDA-LSTM mechanism**

As shown in the Figure 7, this system is for any user who needs recommendations in any particular field. For example, it can be used for a movie recommendation, an Amazon or Gmarket product recommendation or for a crowdfunding project recommendation. Here a user requests for a recommendation item, the relevant data required is fetched. This data includes product feature, comments or reviews on it and its description. After preprocessing this data is passed to the

processing unit where the relevant tasks are performed on the respective input data. In the final step, users are recommended with an optimal item based on their preferences.

Our proposed recommendation system is divided into multiple units or modules i.e. (a) text analysis module based on topic modeling (b) deep learning module that classifies and predicts the topic class of any given document (c) module of credibility assessment and (d) recommendation module.

The theoretical framework of our suggested system is further expanded into a layered view in Figure 8. The system takes data and first it is handed over to the text analysis module that is responsible for topic modeling and relevant parameters estimation. The results in form of learnt topics and parameters are passed to deep learning module which trains the system on labeled data and then the testing on the new or unseen data. For each new document it will predict the topic and classify the product into appropriate class. The next module learns all the features of the data and assess the importance or impact of each feature on product credibility by the devised formula. After this the optimization is performed on all features to get the highly desired features into more power. The final recommended item depends upon the credibility and user's interests.

As there are multiple modules in our system, we have implemented our system in multiple phases. Each phase targets one particular module. At the first stage, we focused on data crawling and its preprocessing. In the second phase, we implemented the text analysis module which is the baseline algorithm for all other modules. All other modules are dependent on the output of text analysis. In this phase, we developed topic modeling approach on the textual data and discovered different topics.

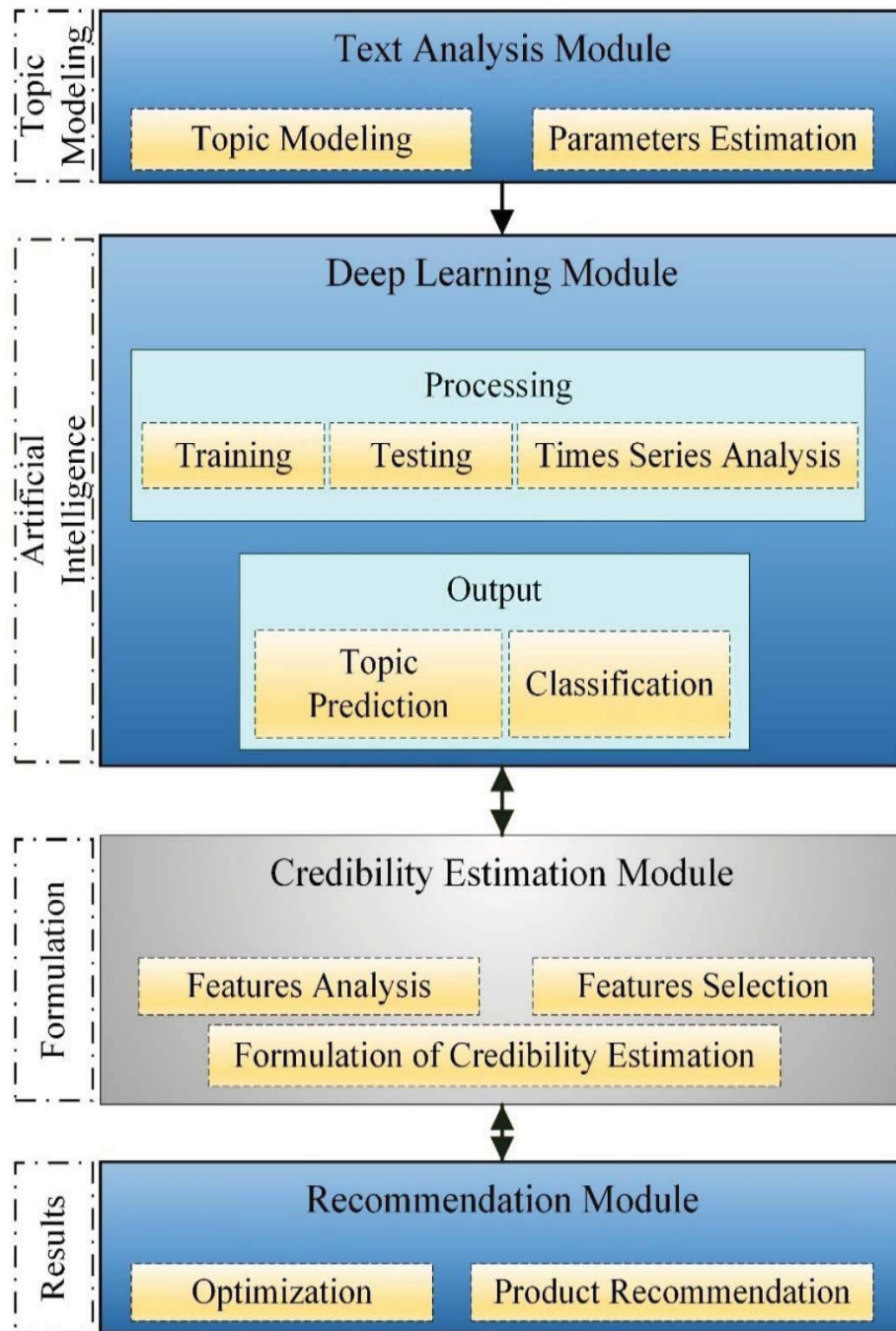


Figure 8: Layer-view for hybrid LDA-LSTM mechanism

In the third phase, the core deep learning module was developed. All the preprocessing required was performed, data was prepared for training and model parameters were learnt. In this phase major classification and prediction tasks were performed. In fourth phase four, we first

formulated a function to estimate the credibility of a product. We used all the features and assessed their impact on the final results. In the final phase, we used optimization algorithms.

### 3.2 Topic Modeling based on LDA

Here, the LDA based topic modeling approach is presented. In section 3.2.1 we present the data preprocessing task. Data preprocessing is a primary part of any NLP task. In section 3.2.2 we present the configurations and parameters of LDA used.

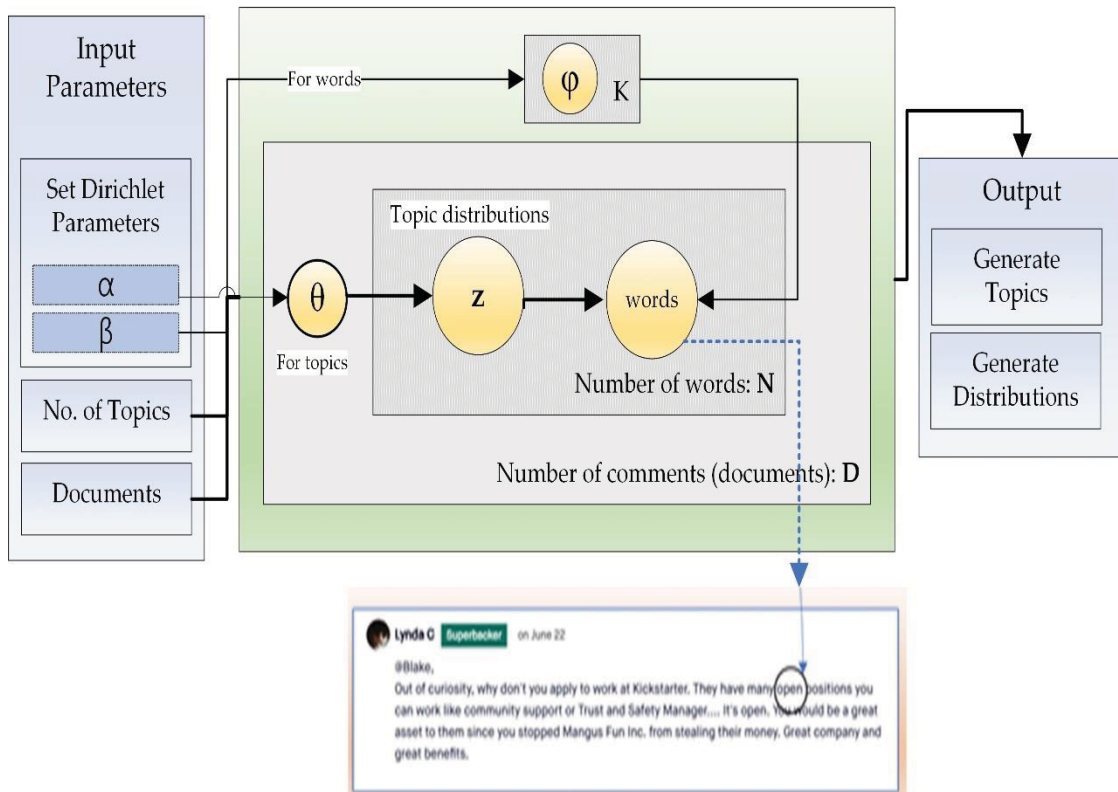


Figure 9: Plate notation of LDA using crowdfunding comments

#### 3.2.1 Data Preprocessing

As data preprocessing is a fundamental task of topic modeling. Before applying LDA, we cleansed the data. Several steps were applied to get the desired form of the data. Data preprocessing is also important because of its impact on the quality and effectiveness of the results.



In this section, we present the LDA based topic modeling approach. In section 3.2.1 we present the LDA parameters required.

Therefore, this unit is responsible for multiple functions. First thing that data preprocessing unit does is to tokenize the text document such as comments or reviews into words. Once the document is tokenized it is passed through cleansing unit. This unit removes all the punctuations, single or double quotes, and URLs from the given documents. Next it is passed through stemming unit. This unit lower cases all the words and convert each word to its root. (e.g. *working* is converted to *work*). Third step for this unit is to remove the stop words. Stop words are the words which are used in any language for grammatical reasons (e.g. *a, an, is* etc.). The preprocessing step leaves us behind with only useful and meaningful words. After this processing the document is ready to be passed to LDA for further processing.

### **3.2.2 LDA Parameters and Configurations**

Here we describe and present the detail of LDA model in terms of its parameters and configurations.

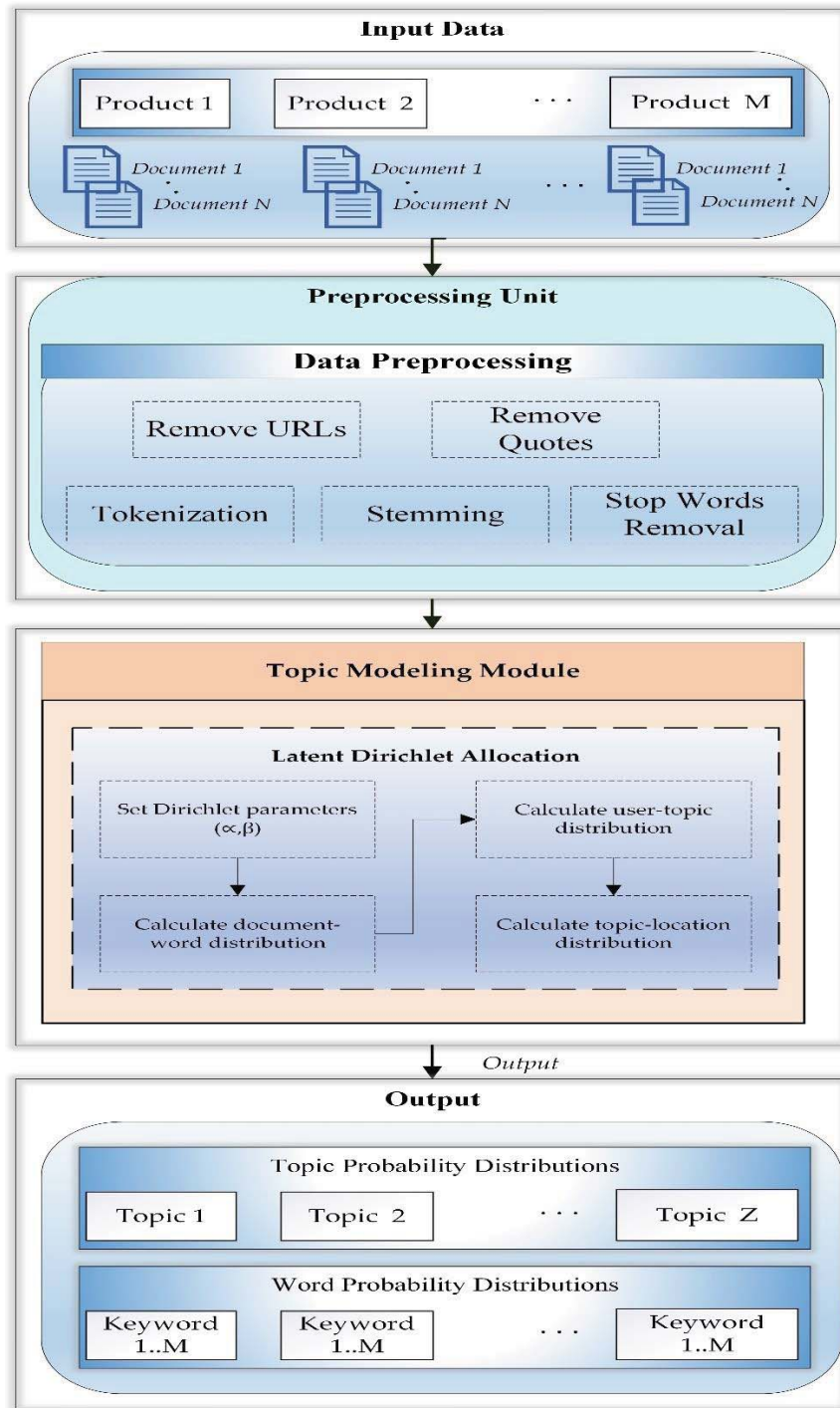


Figure 10: A detailed architectural view of LDA process in our system

As shown in the Figure 10, we use LDA to find the hidden topics within the text data available in form of comments or reviews. One of the key functions of LDA are to set the Dirichlet parameters i.e.,  $\alpha$  and  $\beta$ . Other parameters of LDA are listed in Table 5.

**Table 5: Parameters of LDA**

<b>Model Parameters</b>	<b>Nature</b>	<b>Description</b>
<b>K</b>	Integer (INT) type	Quantity of topics
<b>V</b>	INT type	Dictionary/Vocabulary Size
<b>D</b>	INT type	Quantity of documents
<b>N</b>	INT type	Quantity of words in document d
<b><math>\alpha</math></b>	K-dimensional vector [+ve real]	topic $k$ 's prior weight
<b><math>\beta</math></b>	V-dimensional vector [+ve real]	Word's prior weight in topic $k$
<b><math>\theta</math></b>	Float [0-1]	Probability value
<b>z</b>	N-dimensional vector of integers	Represents the value of topic assignments

The number of topics is represented by  $K$ ,  $V$  represents the vocabulary or the dictionary size,  $D$  denotes total quantity of text documents and  $N$  represents total quantity of words per document. All these parameters are of integer type.

Other parameters include  $\alpha$  which effects the topic distributions per document. If the value of  $\alpha$  is high, it will result in better or uniform distribution of topics in each document. Similarly,  $\beta$  that influences the words distribution per topic. If the value of  $\beta$  is high, it reflects the more uniform distribution of words per topic.

### 3.3 Deep Learning Methods based on RNN-LSTM

We are using a bidirectional LSTM for capturing the context dependencies with respect to time. In a bidirectional LSTM when an input is provided then it analyzed in its natural order as well as inverse order. This is done to capture maximum dependencies within the data. We are using a 128-unit LSTM (bidirectional) for this purpose. The input from preprocessing module is passed to an embedding layer which converts input into 64-bit vector representation. This representation is then processed by the LSTM layer which is then connected to a dense layer. This layer helps to consolidate the LSTM results. This is then connected with the outer layer. The output layer gives probability distribution of output category. The details of this neural network are summarized in following Table. 6.

**Table 6: Neural Network Configurations**

Layer	Type	Configuration
Input Layer	Dense ([Word of bag of embedding size] + Topic Distribution + Topic Embeddings)	Word Embeddings [200 * 1000] + probability distributions [(1000 * 12) + (200 * 12)]

Second Layer	Bidirectional LSTM Layer	128 units
Third Layer	Dense	64 unit
Output Layer	Dense	12 (Output Classes)

### 3.4 Credibility Estimation

This section presents detailed explanation of our credibility estimation module. In this scenario of crowdfunding projects, our aim is to develop an objective function to search and discover the most reliable project to recommend to user.

Trust being an important and crucial factor in any domain, is equally imperative for online social networks and electronic commerce sites to improve customer relationships. There are various recommendation systems built on trust building mechanisms referred as trust-aware recommender systems. These systems use user's trust statements and their personal or private data provided at their SNS profiles. The inclusion of these trust statements or social factors in any recommender system can significantly improve the recommendations quality [120-123].

A highly credible recommendation is a product or item that matches the user defined interests and categories with maximum probability. Also, with lowest probability of elements that can affect the reliability of the product such as communication delays and high price etc.

For example, in case of crowdfunding project recommendation, a credible or authentic recommendation is a project that has highest chances of being delivered on promised time. We define and link a documents' credibility with its estimated authenticity score range. The

authenticity of a document is resultant of various factors primarily from latent patterns of communication. Such factors involve a creator’s profile and his or her social links. It also involves how often creators use their accounts to update or post a comment, that keeps the investors updated and involved in the process which increases the transparency of the process. The other factors involved include most frequent keywords used, promises related to product delivery or rewards delivery, and investors’ sentiments such as how happy or aggressive they appear in the comments they post, etc.

As far as the textual data is concerned in form of comments or reviews, a product can have multiple comments falling into different topic classes. Therefore, the proportion of comments in each topic category is discovered. All the content is divided into three main categories referred as Class A for extremely negative content, Class B for negative content, and Class C for positive content. We have put more focus on the content that shows negative emotions. Therefore, we have assigned two representative classes Class A and Class B to entertain these negative comments. The classification is based on the nature of negative content. The rest of the comments belong to Class C.

The reason behind this arrangement with more emphasis upon negative comments is the underneath cruciality or impact of the negative content on the credibility of a product. Table 7 summarizes the parameters used for authenticity measures with their definitions and notations.

**Table 7. Definitions of the parameters of authenticity**

No.	Authenticity Parameters	Description	Notations
	<b>Content based</b>		

1	Class A	Weightage of comments/reviews in extremely undesired category	$X_w$
2	Class B	It represents the total percentage of the comments in negative category	ClassB
3	Class C	%age of comments in positive category	ClassC
4	Readability score	Measure of content clarity	Rscore
<b>Profile based</b>			
5	Profile picture	It's a binary feature to check if the creator of a project has provided his/her profile picture or not	$Y_w$
6	Total number of social links	This feature represents the total number of external links to other social sites e.g., Facebook of the creator are on his crowdfunding profile	$links_{social}$
7	Delay between posts	It denotes the average time difference between updates, comments, or reviews	$delay_{post}$

Hence, by incorporating all the above-mentioned factors, we have formulated an equation that helps calculate the authenticity of a given project. In order to calculate the authenticity of a

project, it must first fulfil the eligibility criteria given in Equation 1. Once a project passes the eligibility criteria, Equation 2 is used to calculate the authenticity of it. The eligibility criteria is based on the content of a project and partially on the profile associated features.

$$Eligibility_{criteria} = -(X_w + \alpha Y_w) \quad (1)$$

Here,  $X_w$  and  $Y_w$  represents the weights of Class A and existence of a profile picture respectively.  $Y_w$  is given comparatively less weightage as compared with  $X_w$  because of the level of impact asserted by each parameter. The value of  $\alpha$  is set to 0.5.

From the above equation, we define the ranges for both the parameters.

$$X_w = \begin{cases} 0 & \text{if Class A} > 0 \\ 1 & \text{if Class A} \leq 0 \end{cases} \quad (2)$$

Similarly,

$$Y_w = \begin{cases} 0 & \text{if profile picture} = \text{Yes} \\ 1 & \text{if profile picture} = \text{No} \end{cases} \quad (3)$$

Hence from above Equations 2 and 3, we have

$$Eligibility_{product} = \begin{cases} 0 & \text{desirable} \\ < 0 \geq -0.5 & \text{can be considered} \\ < -0.5 & \text{undesirable} \end{cases} \quad (4)$$

Therefore, based on the above Equation 1 and following the conditions in Equation 4 we can list down all the possible scenarios of eligibility in Table 8.

**Table 8. All possible scenarios of project eligibility criteria**

$X_w$	$Y_w$	Explanation	Eligibility  $Eligibility_{product} = -(X_w + \alpha Y_w)$ ( $\alpha=0.5$ )
0	0	No comment in Class A and profile picture exists	0 (desirable)



0	1	No comment in Class A and no profile picture exists	-0.5 (can be considered)
1	0	Comments in Class A exists and profile picture also exists	-1(undesirable)
1	1	Comments in Class A exists and profile picture does not exist	-1.5 (undesirable)

The content in Class A is of very undesired nature as one can sense threats, disbelief, and frustrations in it. That is why this class has been treated separately to mitigate the probability of any unreliable recommendation. For a product or project to be completely reliable, it must not contain any of the comments in this category. That is why, we used this to set our eligibility criteria. The objective function aims at getting the highest percentage of Class C. It also tries to get the highest number of social links of the content creator.

In case of crowdfunding, the trust and confidence of an investor is dependent on the content authenticity and creator's transparency. Therefore, these factors are very crucial for the success of a project. In the above table, the factor  $\text{delay}_{\text{post}}$  is one primary feature of the project representing a creator's patterns of communication such as, his updates and comments.

This feature,  $\text{delay}_{\text{post}}$  can be defined as the average difference or gap between any successive updates or comments of the project creator. It displays creator's participation and communication rate towards development of the project. Due to the cruciality of  $\text{delay}_{\text{post}}$ , the authenticity of a project will be harmed if the communication delay increases. The values of all

the features are normalized between 0 to 1. Here, 0 represents the least authentic item and 1 represents the highly authentic item. In other words, these values depict the trustworthiness of a project. Equation 5 below, well describes this relationship i.e., the higher the authenticity is, the higher the reliability of a project turns out.

$$Authenticity_{document} \propto Credibility_{document} \quad (5)$$

As a result, there are five various levels of credibility a project has. These levels are named as extremely low credibility, low credibility, normal credibility, high credibility, and extremely high credibility. Each credibility level falls into a different degree of authenticity range. The first two levels i.e., extremely low and low credible projects have higher chances of getting forged. If we put it differently, projects having lower credibility levels has highest probability of suffering issues such as non-payments, no communication or lack of communication, non-delivery, and delays in posts by creator in form of updates or comments. Therefore, such projects should not be recommended for investments to backers. On the other side, a project having higher level of credibility i.e., high or extremely high credibility is undoubtedly a desired project to be recommended to backers because it has higher chances of on time delivery and has a balanced communication pattern.

For any recommendation system, the percentage of positive and negative documents is very important as it tells the overall attitude of a user towards a specific product. Therefore, we will consider the following points carefully:

1. For example, for a very basic analysis, we can say that for a product to be trustworthy it must have maximum positive comments and minimum negative comments. Let's consider, if a product has comparatively a large number of negative reviews, it will be

less favorable. Thus, we can say that authenticity is directly proportional to the percentage of positive comments. Therefore,

$$\text{Authenticity} \propto [\text{Class } C_i] \quad (6)$$

In the above Equation 6, Class  $C_i$  refers to the percentage of positive comments.

Similarly,

$$\text{Authenticity} \propto [1/\text{Class } B_i] \quad (7)$$

In the above Equation 7, Class  $B_i$  is the percentage of negative comments.

2. Authenticity strongly gets influenced by the presence of social information such as presence of profile picture, the social networks links, etc. Thus, the more a person is providing information about himself or herself, the more easily the product earns trust. Therefore,

$$\text{Authenticity} \propto [\text{links}_{\text{Social}}] \quad (8)$$

In the above Equation 8,  $\text{links}_{\text{Social}}$  is the number of links a person provides for his/her external social media networks, such as Facebook, twitter etc.

3. The content clarity also plays an important part towards trust building. It means if the document is easy to follow and understand, a user will find more connected and will understand the content without any confusion it helps minimize the confusions and the surety or confidence level increase. Therefore,

$$\text{Authenticity} \propto [1/R_{\text{score}}] \quad (9)$$

In Equation 9,  $R_{\text{score}}$  is the readability score of a document. If  $R_{\text{score}}$  is high, the document is difficult to follow or to understand. The lower the readability score is the higher probability is to understand it quickly.

4. The communication patterns are the key towards trust building. If a communication is consistent and smooth, people will enjoy and will able to put their trust on it. If there is

no communication from the product creator to the concerns of the customers or user, it will make users frustrated and they will start lacking their interests. The delay should be minimum between the posts or response by the creator.

$$\text{Authenticity} \propto [1/\text{delay}_{\text{post}}] \quad (10)$$

In Equation 10,  $\text{delay}_{\text{post}}$  is the average delay between any consecutive posts, comments or updates by the product creator. If the delay is larger, authenticity will be affected negatively. For higher authenticity, the delay should be minimum.

5. Hence, we can summarize the above-mentioned factors as:

$$\text{Authenticity} \propto [\text{Class } C_i, \text{links}_{\text{Social}}] \quad (11)$$

Also,

$$\text{Authenticity} \propto [1/\text{Class } B_i, \text{delay}_{\text{post}}, R_{\text{score}}] \quad (12)$$

By combining Equations (11) and (12), Equation (13) is formulated as below,

$$\text{Authenticity} \propto [\text{Class } C_i, \text{links}_{\text{Social}} / \text{Class } B_i, \text{delay}_{\text{post}}, R_{\text{score}}] \quad (13)$$

6. In the next step, we will derive the final equation of credibility estimation by considering all the above-mentioned relations and factors. Therefore, if we see the factors such as positive and negative comments they are of same type, hence we divide the equation in two parts, the similar type of factors based on their priority are assembled together. Hence, Equation 14, separates the sentiment-based classes.

$$\text{Authenticity} = [\text{Class } C_i / \text{Class } B_i] \quad (14)$$

This factor is only related to the comments or reviews a product have. That is why we have separated it. In order to have higher authenticity, Class  $C_i$  has to be larger than Class  $B_i$ . We have separated other features related to the product or creator into one factor as,

$$\text{Authenticity} = [\text{links}_{\text{social}} / R_{\text{score}} + \text{delay}_{\text{post}}] \quad (15)$$

7. Now, at this step, we will combine all the factors in one place that results into Equation 16 as below,

$$\text{Authenticity}_{\text{document}} = \left[ \sum_{i=1}^n \frac{\text{Class}_{ci}}{\text{Class}_{Bi}} + \left( \frac{\text{links}_{\text{social}}}{R_{\text{score}} + \text{delay}_{\text{post}}} \right) \right] \quad (16)$$

8. At the final step, we apply optimizations and formulate our objective functions. We have both maximization and minimization functions. The maximization function maximized the value of parameters which are most desired. On the other hand, minimization function minimizes the value of parameters that are least desired for an optimal and credible product. We have used PSO as an optimization algorithm to tune the assigned weights to each parameter in a very optimal manner.

Therefore, we can now formulate the credibility estimation in terms of maximization and minimization functions. It results into Equation 17.

$$\text{Credibility}_{\text{product}} = \left[ \sum_{i=1}^n \frac{\max(\text{Class}_{ci})}{\min(\text{Class}_{Bi})} + \left( \frac{\max(\text{links}_{\text{social}})}{\min(R_{\text{score}}) + \min(\text{delay}_{\text{post}})} \right) \right] \quad (17)$$

For the above equation, we can define the value ranges for each parameter as below in Equations 18 to 22.

$$\text{Class}_B = \begin{cases} 0 & \text{if \%age of negative comments} = 0 \\ 1 & \text{if \%age of negative comments} = 100\% \end{cases} \quad (18)$$

$$\text{Class}_C = \begin{cases} 0 & \text{if \%age of positive comments} = 0 \\ 1 & \text{if \%age of positive comments} = 100\% \end{cases} \quad (19)$$

$$\text{links}_{\text{social}} = \begin{cases} 0 & \text{if Number of links} = 0 \\ > 0 \leq 9 & \text{maximum number of links} \end{cases} \quad (20)$$

$$R_{\text{score}} = \begin{cases} \text{near } 1 & \text{easy to understand} \\ \geq 50 \leq 100 & \text{difficult to understand} \end{cases} \quad (21)$$

$$\text{delay}_{\text{post}} = \begin{cases} 0 - 365 & \text{days} \end{cases} \quad (22)$$

Following Table 9, we can define the maximum and minimum ranges of each parameters.

**Table 9. Value ranges for each credibility parameters**

No.	Parameters for Credibility	Maximum Range	Minimum Range
<b>Content based</b>			
1	Class <sub>B</sub>	100	1
2	Class <sub>C</sub>	100	1
3	R <sub>score</sub>	100	1
<b>Profile based</b>			
5	links <sub>social</sub>	9	0
6	delay <sub>post</sub>	365	0

In Figure 11 we have shown the work flow of PSO algorithm. The first and the second step in the PSO algorithm is to randomly initialize the particle velocities for the generated population and to assign its position. The next step is to calculate the fitness of each particle depending on the pre assigned positions and velocities and to obtain the objective function considering both maximization and minimization function. After

which, we make a comparison of the current fitness value to each of the particle's best fitness value, i.e.,  $Y(t)$ .

We update the  $Y(t)$  if the current value is better and then compare  $Y(t)$  with global fitness  $G(t)$ . So, if  $G(t)$  is better than  $Y(t)$ ,  $Y(t)$  is updated to  $G(t)$  or else it is kept the same. In every iteration, the velocity and position for each particle is updated to its best fitness value.

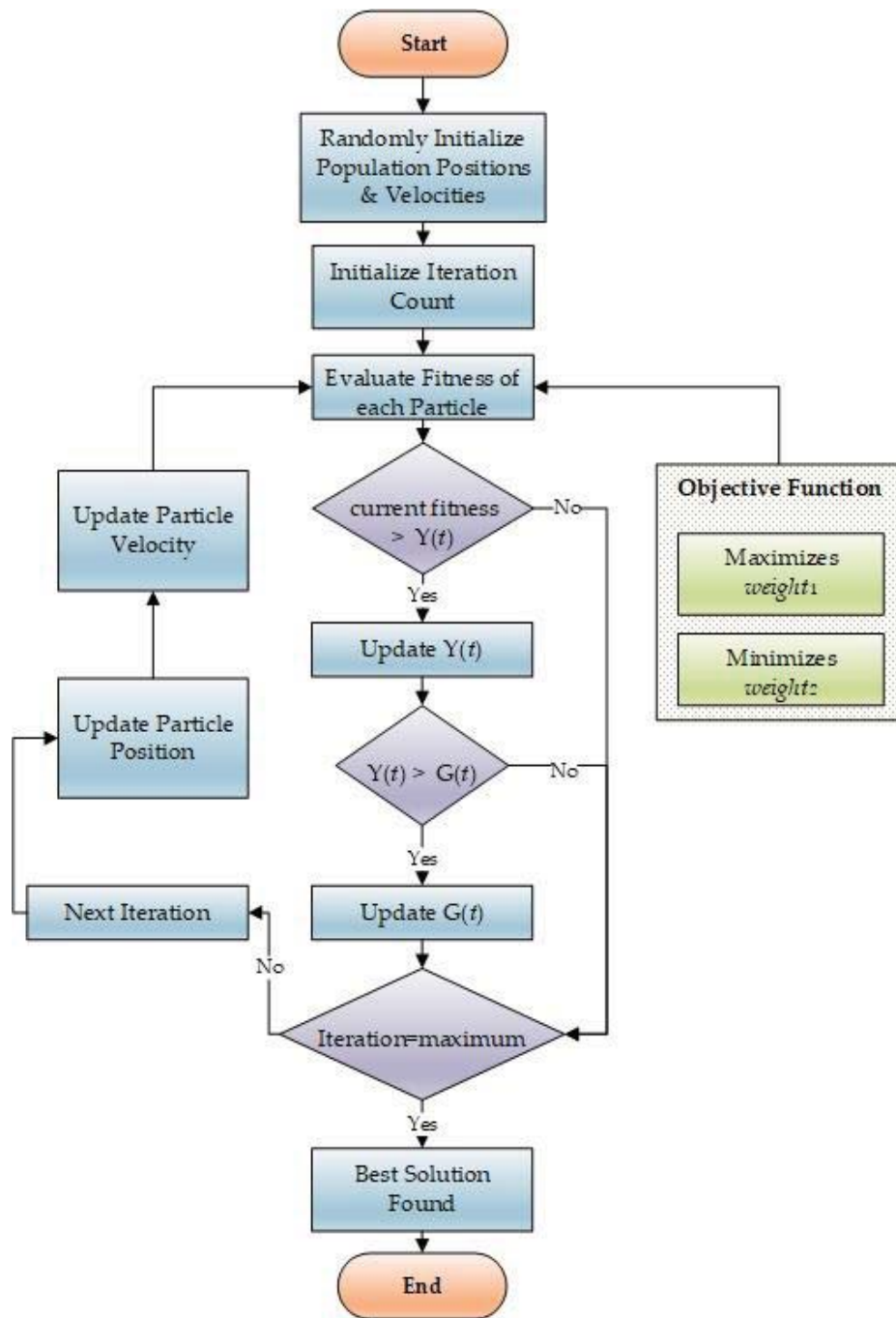


Figure 11: Flow chart for PSO



### 3.5 Overall Structure of the Proposed System

We have proposed a hybrid of LDA and LSTM. Here the strengths of both the techniques are merged together i.e., LDA is capable of effective interpretations of the data and LSTM is capable of keeping track of temporal dependencies in an efficient manner. In this respect, we propose a model that takes care of all the strong points of baseline models in a way that LSTM models topic sequences represented as  $p(z_t|z_{t-1}, z_{t-2}, \dots, z_1)$ . These are learnt through LDA. It also models word emissions represented as  $p(w_i | z_i)$  by using multinomial-Dirichlet parameter that is similar to LDA. The LSTM module takes word embeddings plus topic distributions of the current document along with the dynamic embeddings.

Henceforth, the hybrid model can be summarized in the following Equation 23:

$$\log p(w) = \log \sum_{z_1}^T \prod_T p(w_t | z_t) p(U_t | U_{t-1}; LSTM) * p(z_d, t | z_{d(t-1)}, z_{d(t-2)}, \dots, z_{d1}; LSTM) \quad (23)$$

Here,  $p(z_d, t | z_{d, 1:t-1}; LSTM)$  represents the probability of generating a topic for next word in a document given topics of former words. The probability of topic embedding is represented as  $p(U_t | U_{t-1})$  before normalization.  $U_t$  is the probability of a topic embedding at the current time step  $t$ , and  $U_{t-1}$  is the probability of topic embedding at former time step  $t - 1$ . To model this part of the equation, a Gaussian prior is added with zero value of mean and a variance of  $\alpha 2.0$ . It sets threshold for the dynamic topic embedding vectors to stop them getting too big. So, Equation 24,

$$p((U_t | U_{t-1}) \propto N(U_{t-1}, \alpha 2 t - 1) * N(0, \alpha 2 0 I) \quad (24)$$

The complete architecture and process can be illustrated as in Figure 12.

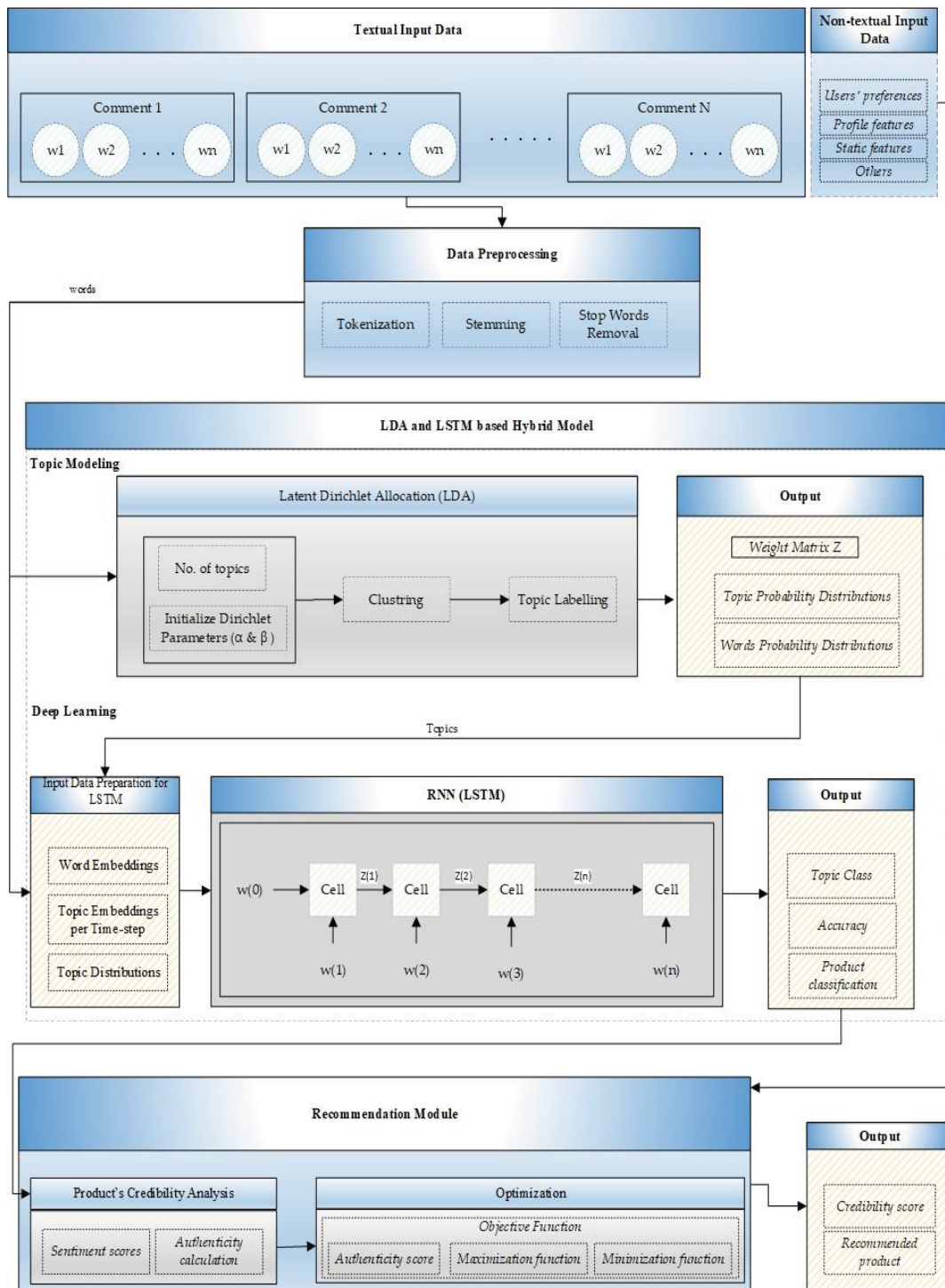


Figure 12: Architecture of proposed hybrid approach

### 3.5.1 Input Data

The input data is of two types, i.e. textual and non-textual data. The textual data can be a comment or a review on any product. This data is represented as a sequence of words. Topic modeling is applied on this textual data to extract the prominent topics. The other data referred as non-textual data includes user's preferences; product static features such as location and category; and profile features of the product creators such as their profile pictures, number of social links, number of posts by them and time delay between their posts. This data is used as input in the product recommendation process.

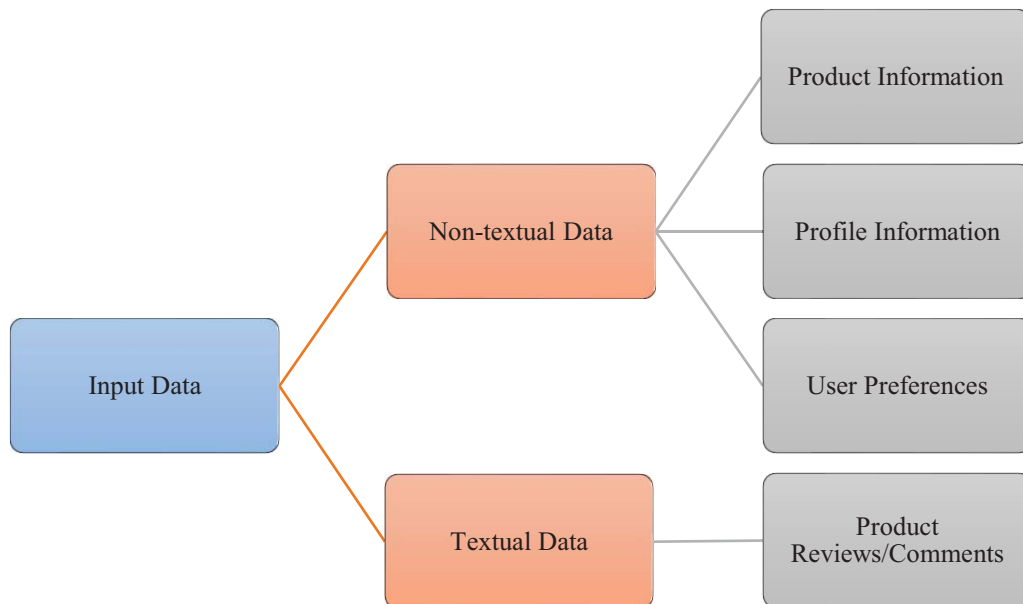


Figure 13. Input data explanation

### 3.5.2 Data Pre-processing

This unit is responsible for multiple functions. First thing that data preprocessing unit does is to tokenize the comments into words. Once comment is tokenized it is passed through cleansing unit. This unit removes all the punctuations from the words. Next it is passed through stemming unit. This unit lower cases all the words and convert each word to its root. (e.g. working is converted to work). Third step for this unit is to filter the comment from stop words. Stop words

are the words which are used in any language for grammatical reasons (e.g. a, an, is etc.). After this processing comment is passed to LDA for further working.

There is a separate pipeline in preprocessing unit which is responsible for preprocessing data for LSTM. In this pipeline the comment is first tokenized, then is lower cased and then all punctuations are removed. The difference from earlier flow is that here comment is converted into vector encoding using Bag-of-words. These vector representations are then used by LSTM to further process the data.

### **3.5.3 LDA and LSTM based Hybrid Model**

The preprocessed data is passed to the hybrid module which is responsible for primary processing of the data. Here, data is first handed over to the topic modeling process where LDA is applied on it. The number of topics and Dirichlet parameters are initiated. LDA generates clusters of the words that have highest similarity. Then we label those clusters into meaningful topics. Therefore, after LDA we have topic distributions representing the probability of a topic in a document and word distributions representing probability of a word in a topic. These probability distributions are then prepared as an LSTM input. For LSTM, the word embeddings and topic embeddings are also generated. These embeddings against each new input review/comment are trained in LSTM network. The topic classes are distributed in three basic classes of sentiments i.e., positive sentiments, negative sentiments and neutral. Therefore, the percentage of each topic class is calculated and assigned a sentiment class accordingly.

### **3.5.4 Recommendations Module**

The primary highlight of this module is that it recommends products based on their credibility. It means this module is responsible for the estimation of a product's authenticity or trustworthiness. It also optimizes the results by incorporating the non-textual features as well.

The authenticity of the product is based on multiple factors such as the product creator's profile, his/her communication patterns and discussion trends of the users. We tried to find the optimal relationship of all these factors with product authenticity. We measured the impact of each factor on product's credibility by analyzing and comparing with the previous studies. Based on the relationship of these factors with authenticity, we have formulated an equation, which calculates the authenticity of a product. Then these authenticity scores are divided into different credibility levels. A product with highest credibility within a user preferred category, will be recommended.

This model has following contributions:

1. The LSTM module takes word embeddings plus topic distributions of the current document along with the dynamic embeddings.
2. For each candidate item, authenticity/credibility is measured based on our implemented optimization algorithm by using Equation 25.

$$Authenticity_{document} = \left[ \sum_{i=1}^n \frac{Class_{ci}}{Class_{Bi}} + \left( \frac{links_{social}}{R_{score} + delay_{post}} \right) \right] \quad (25)$$

3. The use of dynamic and static topic embeddings enables us to save both temporal and contextual dependencies.
4. It posses the power to accurately predict the future trends in discussions.

### 3.6 Structural Details of the Hybrid Model

Here, we will discuss the complete structure of the hybrid model. The proposed model uses LSTM for modeling the sequences of topics given as  $p(z_t|z_{t-1}, z_{t-2}, \dots, z_1)$ . The sequences of the words are modeled by using LDA given as  $p(w_i|z_i)$ . And additionally, we feed our LSTM network with  $p(U_t | U_{t-1})$ , to keep track of dynamic nature of topic assignment to a specific word.

Table 10 presents the generative algorithm for the model where we use topics  $K$ ,  $V$  as dictionary size,  $D$  as collection of the documents and  $N$  as number of words in each document.

**Table 10. Algorithm for our proposed hybrid model**

<b>Algorithm</b>
<p>Start</p> <ol style="list-style-type: none"> <li>1. for <math>k = 1 \rightarrow K</math> (for topics)           <ol style="list-style-type: none"> <li>select a topic <math>\phi_k \sim \text{Dirichlet}(\beta)</math></li> </ol> </li> <li>2. for <math>d = 1 \rightarrow D</math> (for documents)           <ol style="list-style-type: none"> <li>a) <math>s_0 = 0</math>, Initialize the LSTM</li> <li>b) for each position <math>x</math> in the document               <ol style="list-style-type: none"> <li>i. Choose a topic <math>z_{d,x} \sim \text{Multinomial}(\Theta)</math></li> <li>ii. Choose a word <math>w \sim \text{Multinomial}(\Phi_{z_{d,x}})</math></li> </ol> </li> <li>c) for words at <math>t = 1 \rightarrow N</math> <ol style="list-style-type: none"> <li>update LSTM as <math>s_t = \text{LSTM}(z_{d,t-1}, \text{previous state})</math></li> <li style="text-align: center;">LSTM update as <math>s_{t1} = \text{LSTM}(U_t   U_t - 1)</math></li> <li>Calculate the value of <math>\theta = \text{softmax}_K(W + b)</math></li> <li>Select a topic (<math>z_{d,t}</math>) from <math>\theta</math></li> <li>Select word (<math>w_{d,t}</math>) from <math>\phi_{z_{d,t}}</math></li> </ol> </li> </ol> </li> </ol>

# Chapter 4: Crowdfunding Project Recommendations: An Example Application

The tremendous increase in the data availability over the past few years has resulted in the emergence of different techniques and tools which help us understand and analyze the available online data in the best way possible. Data is everywhere, in different forms e.g., text, audio, video etc. Primarily, almost every other site gives its customers a right to leave their comments or reviews on their products so that they can reevaluate and improvise their experiences.

There are numerous factors that led us chose crowdfunding as an example system, application or tool for our proposed system.

Crowdfunding sites are gaining popularity at an accelerated rate since past few decades. Analysis of user's data on crowdfunding sites can be helpful in many ways. It can help improve the overall experience of the users and can help them take decisions.

In short, we can say that crowdfunding appeared as a novel or advanced procedure that magnificently attracts significant crowd to invest in new ideas and projects. It also seems to face the disputes and difficulties when it comes to ensure the accountability check, regulations of the laws, ethics supervision, and right handling of the funds. The concept of crowdfunding where brings authority and empowers everyone to take a part in, it raises few concerns at the very same time.

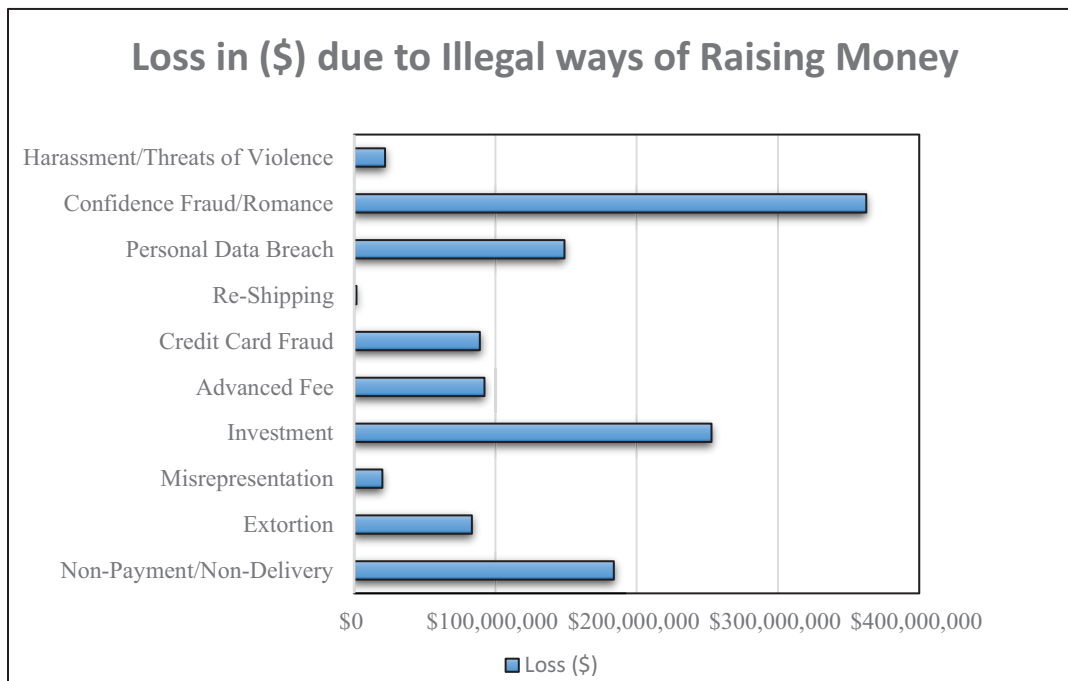


Figure 14. Financial loss due to cyber scams in 2018 according to Internet Crime Report

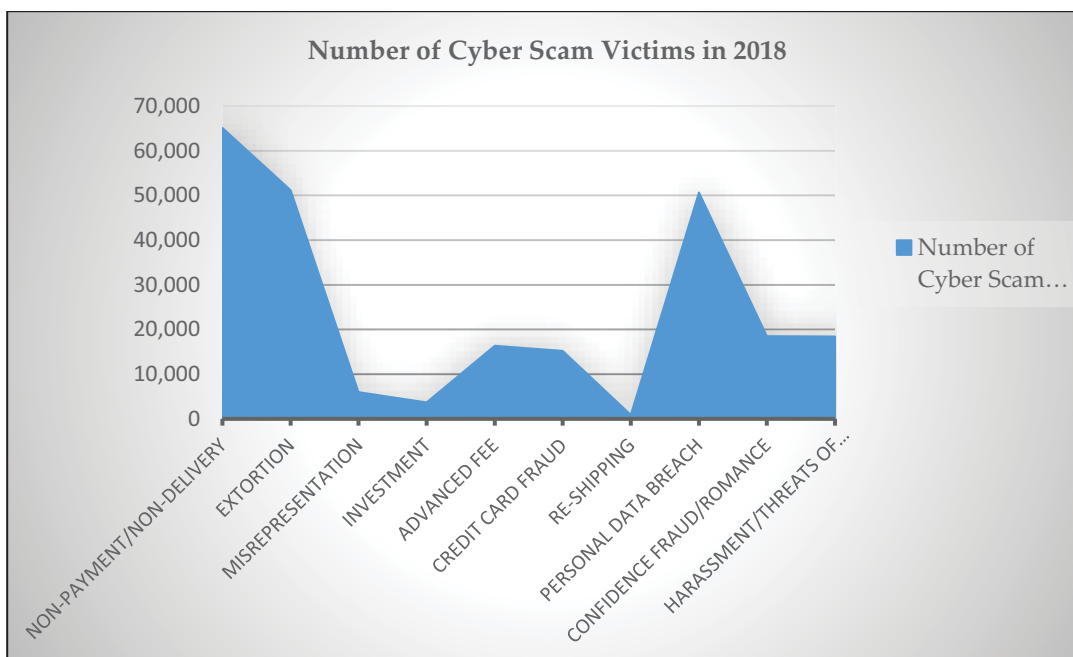


Figure 15. The cyber scams victims in 2018 according to the Internet Crime Report.

There are multiple ways through which fraudsters can raise money illegally. As shown in the Figure 14 and 15, the most common and widely used methods that helped fraudsters to earn huge



amounts include personal data breach, confidence fraud, investments and non-payments or non-delivery.

Therefore, timely identification of such cases where any sort of trust is being harmed is absolutely necessary. There is a dire need of a system for the investors that can recommend them projects that are safe to be invested in.

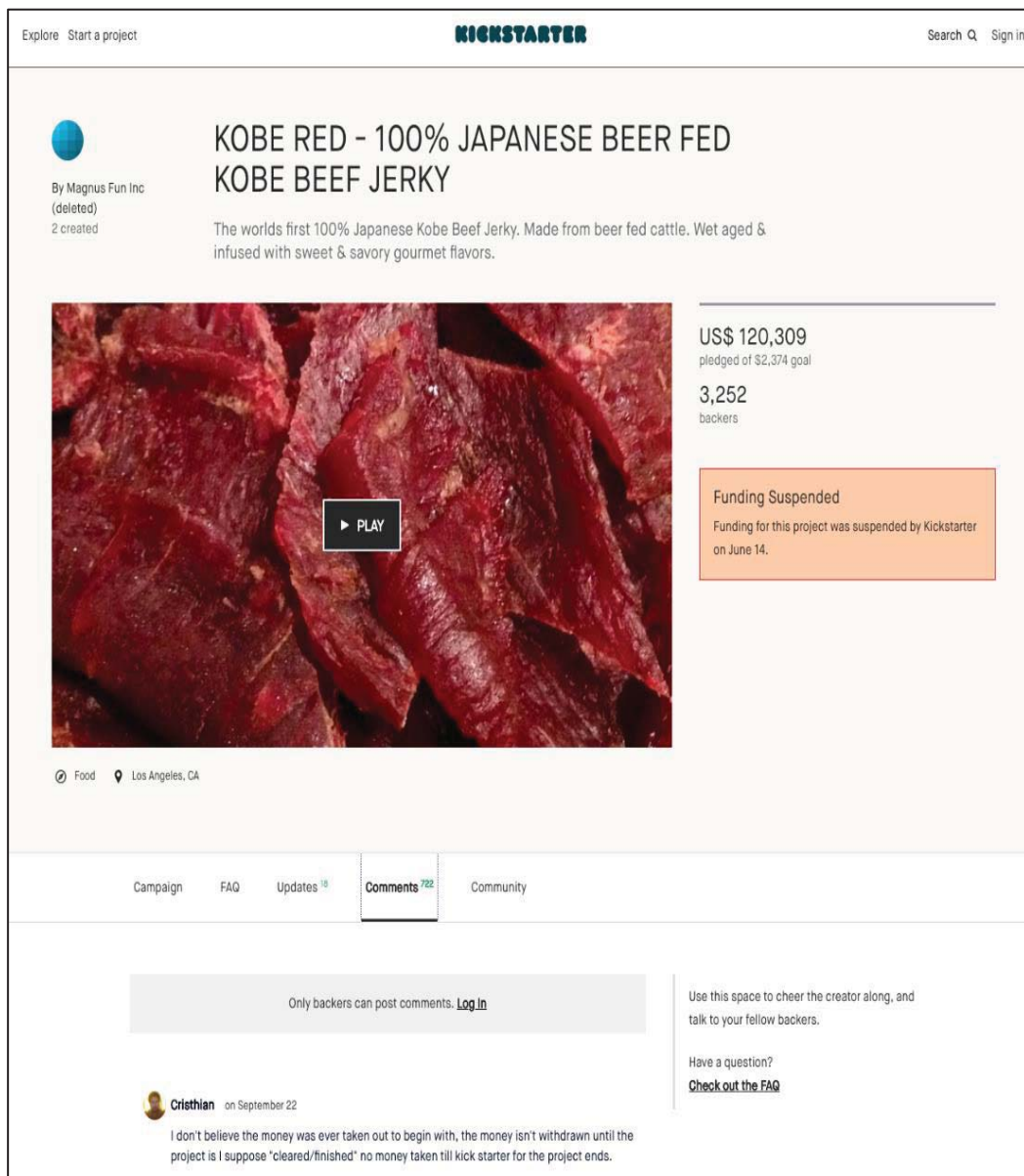


Figure 16. An example of a scam campaign on Kickstarter

Figure 16, is a screenshot of a Kickstarter campaign's home page. It is showing the primary sections of a campaign, status of its funding i.e., suspended in here in the example case. It also has the title of the project, the amount it has raised and the number of backers, etc.

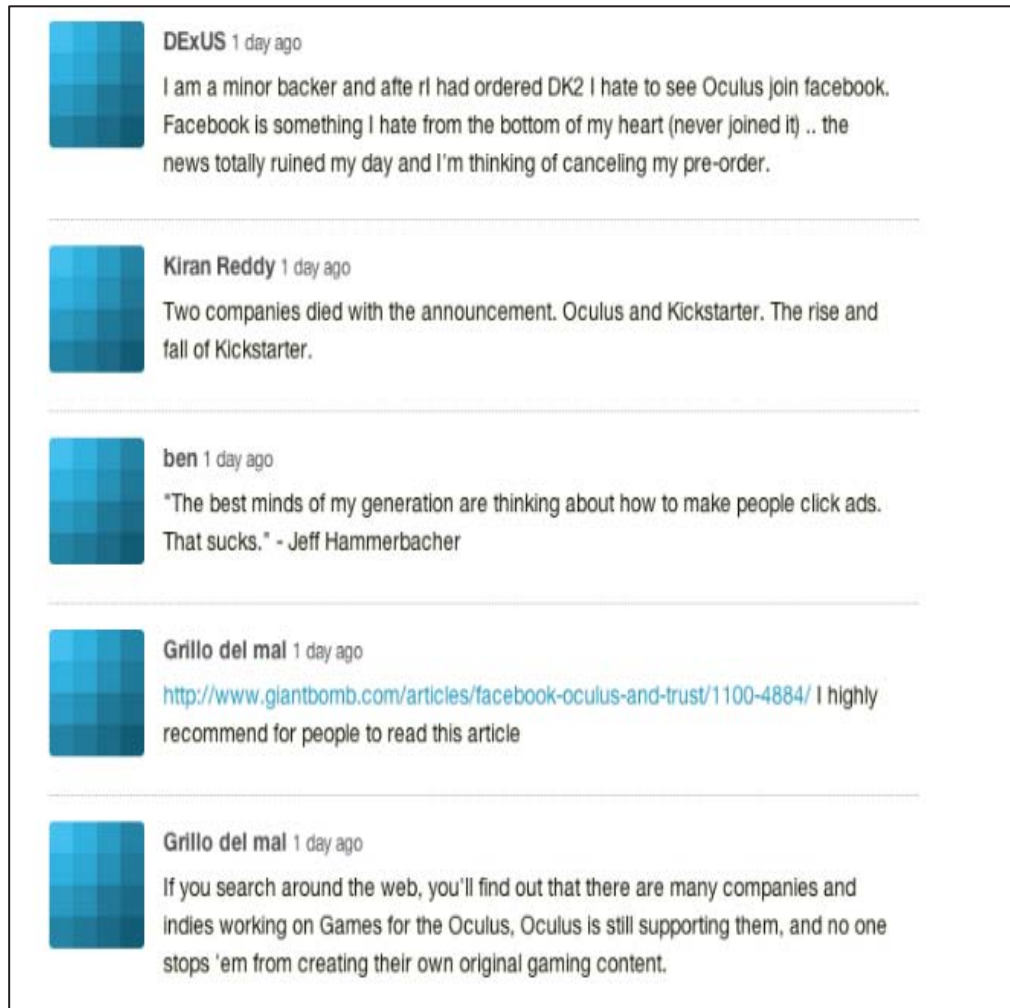


Figure 17. Comments on a Kickstarter Campaign

Figure 17 is the screenshot of a project's comments from Kickstarter. We collected comments for both scam and non-scam categories of projects to have a good representation of the ground truth data.

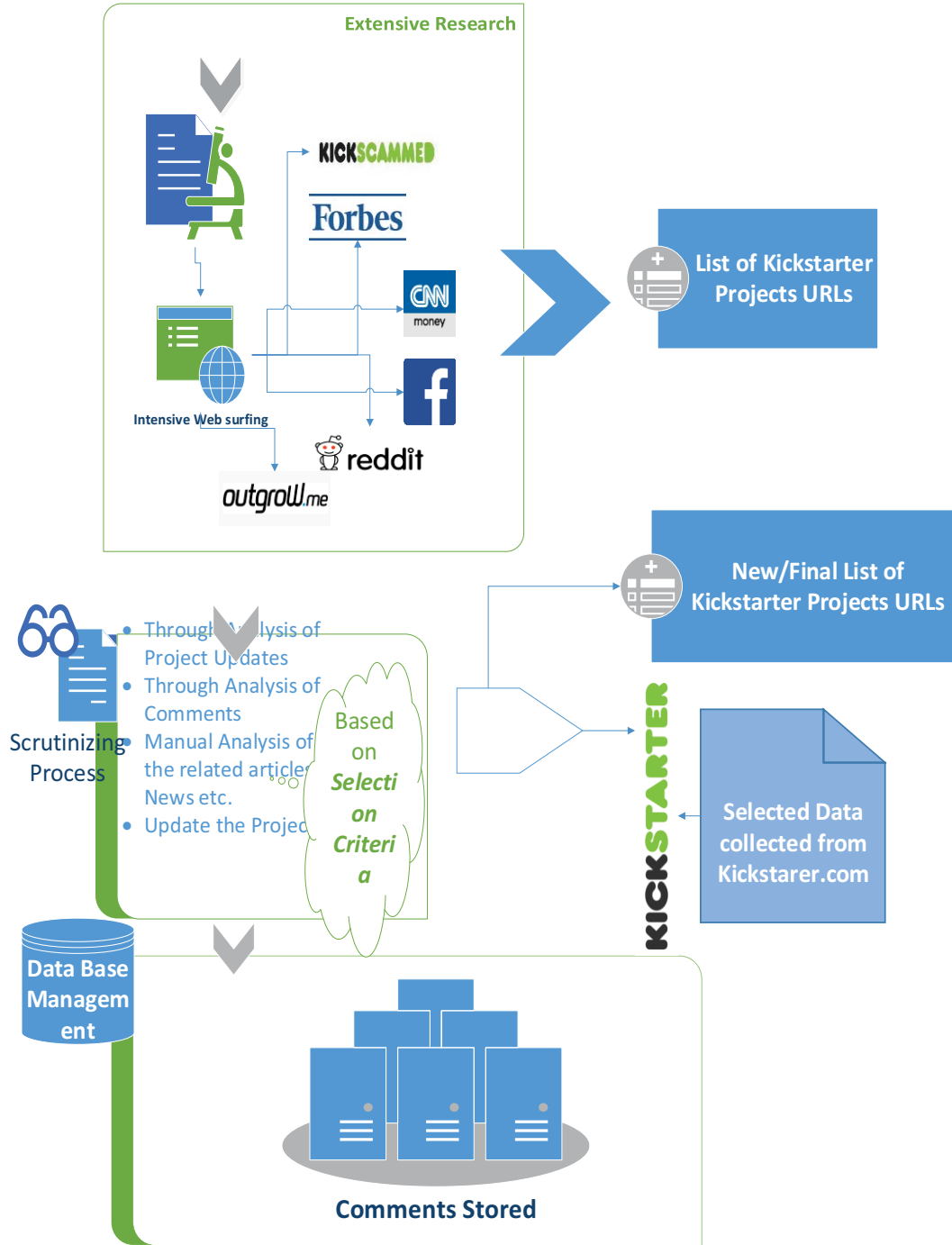


Figure 18. Data collection and selection process

## 4.1 Experimental Data

As mentioned earlier, we targeted comments section of each campaign and collected the comments. All the comments are stored in chronological order.

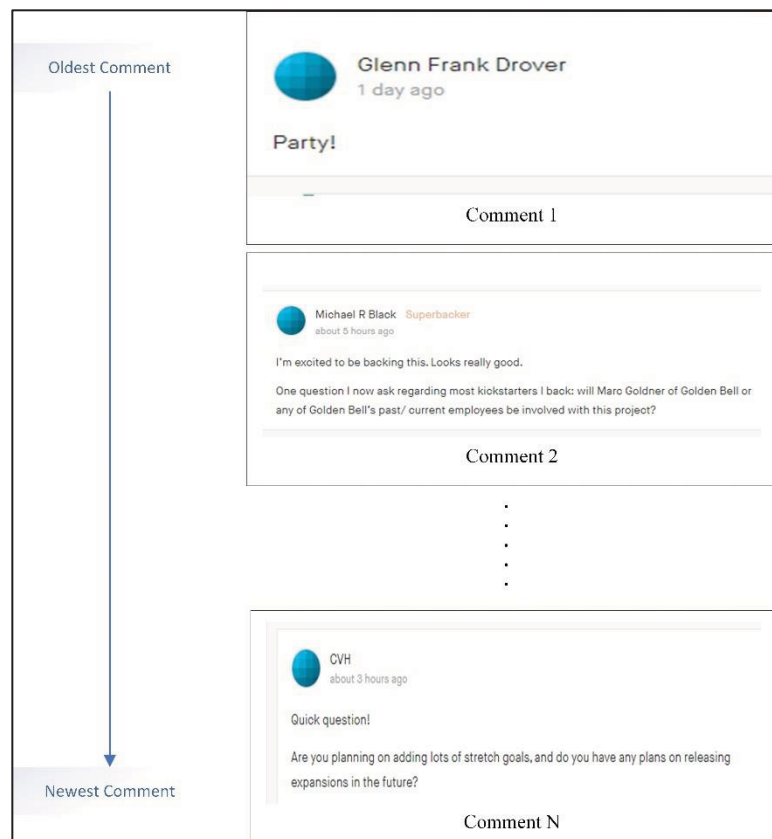


Figure 19. Chronological order view for stored comments

As shown in the Figure 19, the very first comment is Comment 1 and so on. This way we are able to analyze the changing patterns of the discussion trends over time. The total number of comments recorded are 6,45,251. The maximum length considered for each comment is 200 words. After some filtering process on the comments, 5,04,184 comments are left. In total we have 600 kickstarter campaigns that include both the potentially scam and non-scam cases. Each project has 841 comments on average. Table 11 presents the summary of training and test data used for the experiments.

The 70% of this data was used for training and for testing 30% data was used.

Table 11. Details of the implementation and experimental environment

Data Characteristics	Specifications
Total number of projects	600
Total number of comments (before filtering)	6,45,251
Total number of comments (after filtering)	5,04,184
Comments per project (Average)	841
Training data	70%
Test data	30%

## 4.2 Model (LSTM-LDA) Training

In this section we prepare the hybrid model for training. First LDA is applied on the collected comments after the preprocessing. We have comments prepared as bag of words that are passed to LDA for topics discovery. LDA clusters the words based on the similarity and relatedness. LDA generates words probability distributions and topics probability distributions.

These distributions are then converted to words and topic embeddings respectively for LSTM training. Our model uses both the words and topic embeddings unlike the previous approaches. The topic embeddings generated are used at per time stamp that makes the model more efficient. It learns the changes happening in the discussion trends which make it more effective to predict the next topic appropriately.

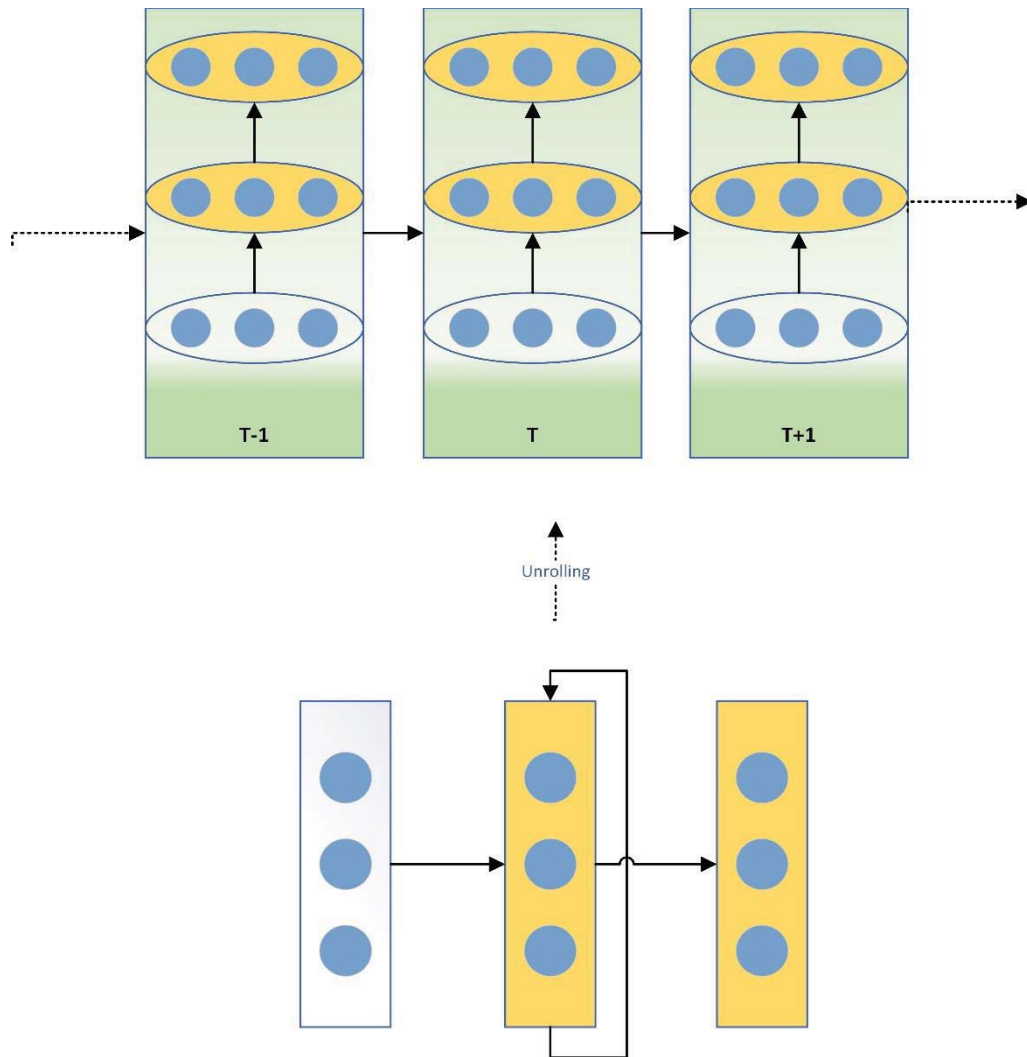


Figure 20. Unrolling view for RNN's basic cell

Each cell of LSTM takes a comment word by word along with the generated distributions and embeddings. Each cell results into one topic class and at end we have the final topic class of the comment. All the comments are processed this way for each project.

After all the comments are passed through LDA, the average of topic classes is taken and based on the inclination of the comments towards negative or positive topics classes, the project is classified into potentially scam or non-scam category.

We achieve a comprehensive evaluation of our joint model by comparing it with traditional or deep-topic modeling approaches. We performed the experiment on the comments collected from Kickstarter. The training corpus is used to extract the latent topics in the comments data. This classifies the comments into different clusters and each cluster represents one particular theme or topic.

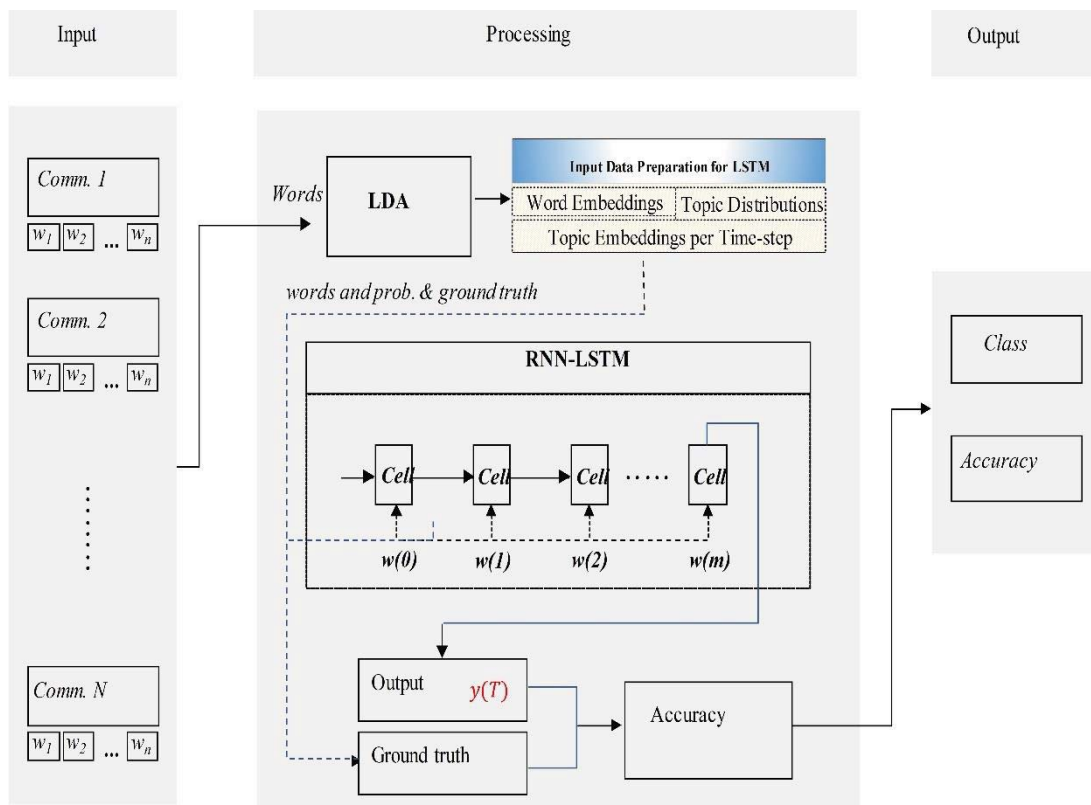


Figure 21. Configuration Diagram for LDA

Table 12: LSTM model configurations

Parameters	Characteristics
Basic data input type	Project Comments

Data format	Project Description, Project Comments, Social Media Links, Profile Picture
Logs start date	2009
Logs end date	2018
Total records	600 Projects
Records span	9 Years
Level of aggregation	Days
Training data	70 %
Testing data	30 %

Table 13 presents the detailed explanation of our LSTM model. The parameters and their types or values are presented. We have used LSTM as an RNN. The input layer takes comments along with the word and topic distributions which are converted to embeddings in the next layer. This layer contains 214,000 neurons responsible to process the input. The next layer contains 12 neurons, which are responsible for the topic categories. We have tested our model on different learning rates.

**Table 13: LSTM model configurations**

Parameters	Characteristics
RNN Type	LSTM



Input Layer	Comment + Word Distribution + Topic Distribution
First Layer	$1 * [(200 * 1000) + (1000 * 12) + (200 * 12)] =$ 214,000 Neurons
Output Layer	12 Neurons (12 Categories)
Hidden Layers	1 (Bidirectional LSTM) + 2 Dense
Learning Rate	0.1, 0.01, 0.001 ,0.005 ,0.0001, 0.00001

After careful analysis of all the clusters, we performed post processing and removed duplicated or meaningless clusters. The remaining topics are then labeled as Topic<sub>0</sub>, Topic<sub>1</sub>, Topic<sub>2</sub>, Topic<sub>3</sub>, Topic<sub>4</sub>, Topic<sub>5</sub>, Topic<sub>6</sub>, Topic<sub>7</sub>, Topic<sub>8</sub>, Topic<sub>9</sub>, Topic<sub>10</sub>, and Topic<sub>11</sub>. These are total 12 topic classes where each class represents one type of discussion in the comments. LSTM uses comments and their labeled topic class for training. Table 14 presents the discovered topics along their detailed explanation.

**Table 14. Identified topics classes after LDA analysis**

Topics	Data	Explanation
Topic <sub>0</sub>	Waiting for rewards	Fulfil, rewards, waiting
Topic <sub>1</sub>	Asking for refunds	Fulfil, refunds, money, creator, project
Topic <sub>2</sub>	Waiting for an update/reward	Waiting, money, refund, update, doesn't, received

<b>Topic<sub>3</sub></b>	Reporting or taking legit actions against it	Attorney, Kickstarter, project, report, actions, response, state, legal
<b>Topic<sub>4</sub></b>	Product never received	Never, still, product, received, mine
<b>Topic<sub>5</sub></b>	- Showing anger or disappointment	Fraudster, what, why, legally
<b>Topic<sub>6</sub></b>	- No communication/confused	Still, no, what
<b>Topic<sub>7</sub></b>	- Product shipment	Product, shipped, wedge, idea
<b>Topic<sub>8</sub></b>	- Product description	Brewer, cup, drink, work, lid, router, device
<b>Topic<sub>9</sub></b>	- Product's working status	Apps, device, great, support, ads
<b>Topic<sub>10</sub></b>	- Product received	Cards, today, mine, received, loved
<b>Topic<sub>11</sub></b>	- Showing excitement	Pledge, received, mine, cards, deck, decks, loving, loved, great

As shown in the Figure 22, we categorized the topic classes into three primary classes of the sentiments i.e., positive, negative, and neutral comments. This helped us to simplify the results and also in the credibility estimation process. Here, Topic<sub>0</sub> is referred as Topic\_0, Topic<sub>1</sub> is referred as Topic\_1, Topic<sub>2</sub> is referred as Topic\_2, Topic<sub>3</sub> is referred as Topic\_3, Topic<sub>4</sub> is referred as Topic\_4, Topic<sub>5</sub> is referred as Topic\_5, Topic<sub>6</sub> is referred as Topic\_6, Topic<sub>7</sub> is referred as Topic\_7, Topic<sub>8</sub> is referred as Topic\_8, Topic<sub>9</sub> is referred as Topic\_9, Topic<sub>10</sub> is referred as Topic\_10, and Topic<sub>11</sub> is referred as Topic\_11.

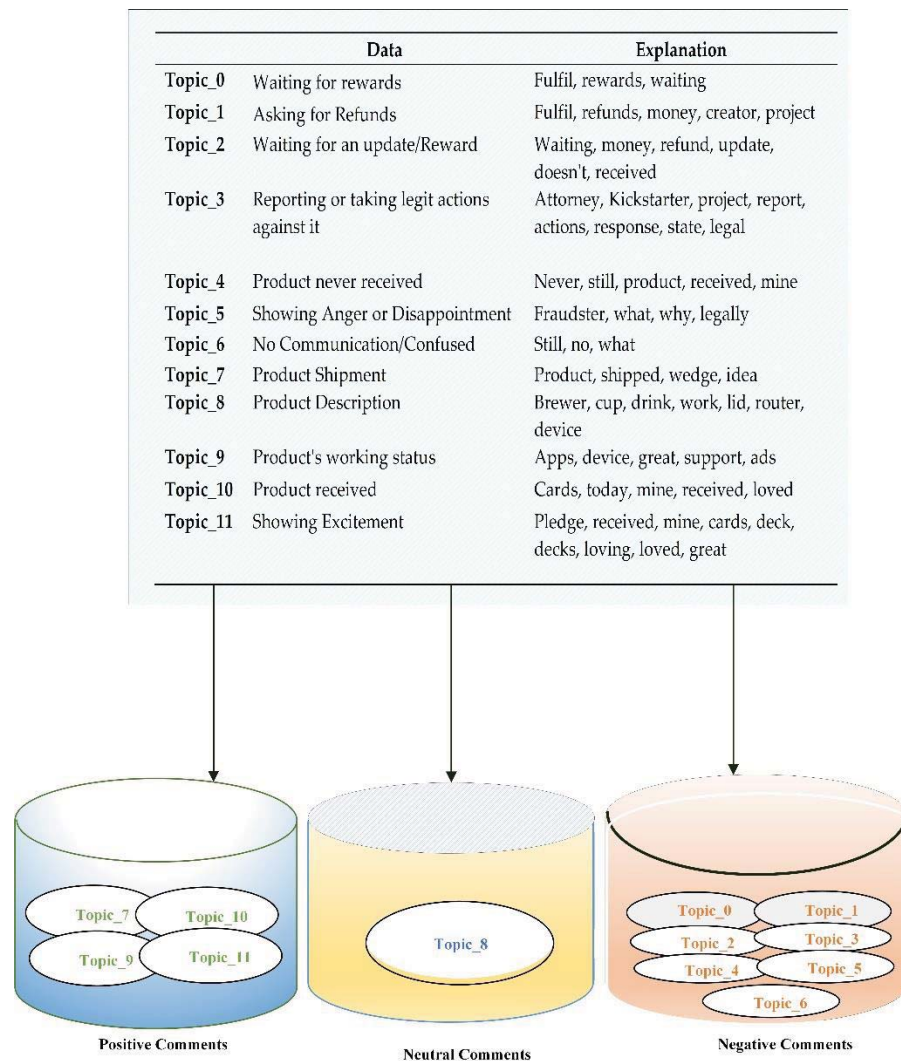
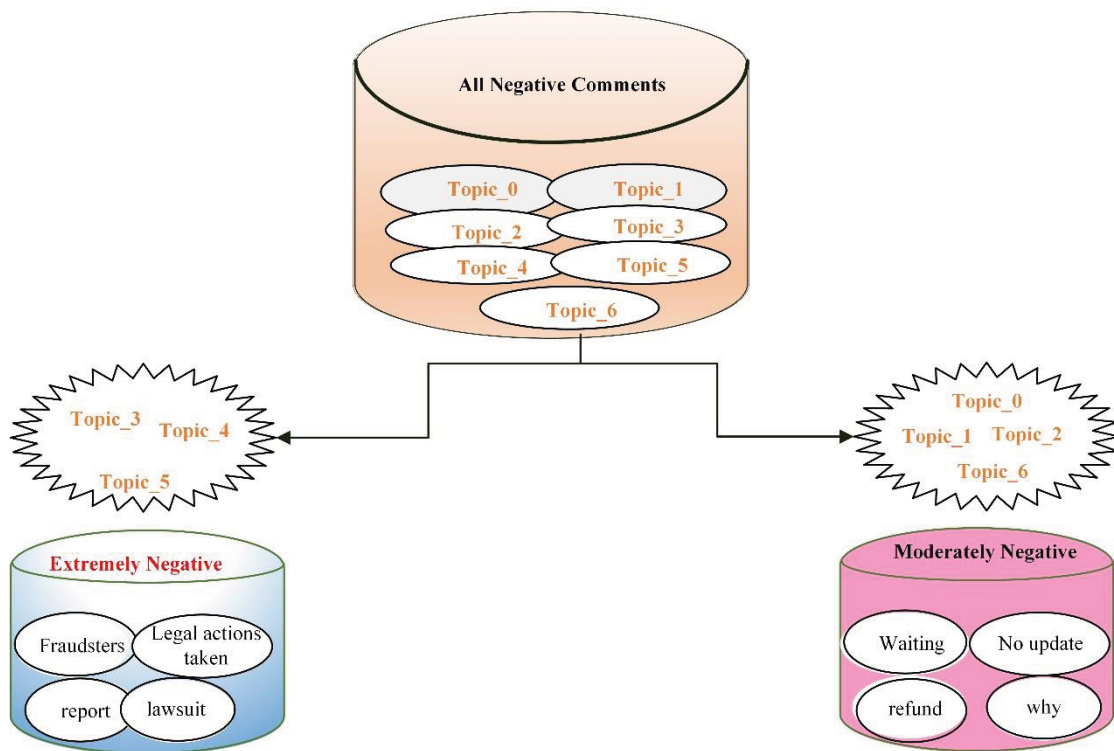


Figure 22. Topic classes categorization

The comments falling into Topic\_0 to Topic\_6 were categorized into negative class of emotions as all these comments reflect negative and aggressive emotions of the investors. Topic\_7 to Topic\_11, cover the comments reflecting positive and neutral emotions of the investors. The first 7 topic classes in the above Figure 22, reflect not so happy attitude of the investors, they are either in a waiting state or have already being disappointed in the progress of the project. Therefore, these classes are very critical in case of assessing the credibility of a project. More comments into these classes means, the project is losing the trust factor. In short, the first 7 topic classes i.e., Topic<sub>0</sub> till Topic<sub>6</sub> are taken as relatively unreliable or doubtful, while the projects in the other categories i.e., Topic<sub>7</sub> till Topic<sub>11</sub> are taken as relatively more reliable and trustworthy.



**Figure 23. Negative topic classes categorization**

Figure 23 shows the further classification of the negative comments related topic classes. As the negative comments in this category are not all at the same severity level. Some comments are extremely negative which cannot be accepted if someone is looking for a reliable project. For

example, the comments where people are already talking about the legal actions they have taken against the project, are very important as the projects with such comments should not be considered. That is why we have further classified these comments into extremely negative and moderately negative comments.

Our system is composed of multiple parts and each component serves a specific purpose. There is a separate pipeline in preprocessing unit which is responsible for preprocessing data for LSTM. In this pipeline the comment is first tokenized, then is lower cased and then all punctuations are removed. The difference from earlier flow is that here comment is converted into vector encoding using Bag-of-words. These vector representations are then used by LSTM to further process the data.

### **4.3 Project Integration**

In context of complete project categorization, the new comment is now processed. Now we collectively process all the comment on the project till now. Each comment now has an attached topic with it. We have already categorized the topics into positive, negative, and Neutral. Now we total and calculate probabilities for each class from the project comments.

### **4.4 Objective Function Formalization for the Optimal Project**

#### **Recommendations**

In this scenario, an objective function aims to search and select a project that is close to user preferred choices and maximum authenticity. An optimal project would be a project that has high credibility or extremely high credibility. There are different credibility levels for a project as shown in Table 15. Therefore, we have to design an objective function in a manner to fulfil the following criteria:

- It must maximize the credibility levels with highest authenticity scores. For example, it must maximize the weights associated with positive topic classes i.e., Topic<sub>7</sub>, Topic<sub>9</sub>, Topic<sub>10</sub>, and Topic<sub>11</sub>
- It must maximize the preferences of the users.
- It must minimize the credibility levels with lowest authenticity scores. For example, it must minimize the weights associated with negative topic classes i.e., Topic<sub>0</sub>, Topic<sub>1</sub>, Topic<sub>2</sub>, Topic<sub>3</sub>, Topic<sub>4</sub>, Topic<sub>5</sub>, and Topic<sub>6</sub>.

Hence, we can formulate the above requirements as following:

$$weight_1 = \alpha_1(Topic_7) + \beta_1(Topic_9) + \delta_1(Topic_{10}) + \gamma_1(Topic_{11}) + \omega_1(UsersPreferences) \quad (26)$$

$$weight_2 = \phi(Topic_0) + \phi_1(Topic_1) + \phi_2(Topic_2) + \phi_3(Topic_3) + \phi_4(Topic_4) + \phi_5(Topic_5) + \phi_6(Topic_6) \quad (27)$$

Here, in Equation 26,  $\alpha_1$  is the weight given to the class Topic<sub>7</sub>, i.e., product shipment.  $\beta_1$  is the weight associated with class Topic<sub>9</sub>, i.e., product working status.  $\delta_1$  is the weight given to the class Topic<sub>10</sub>, i.e., product received.  $\gamma_1$  is the weight given to the class Topic<sub>11</sub>, i.e., showing excitement and  $\omega_1$  is the weight assigned to the user's preferences. Similarly, in Equation 27,  $\phi$  is the weight given to the class Topic<sub>0</sub> i.e., waiting for rewards.  $\phi_1$  is the weight given to the class Topic<sub>1</sub> i.e., asking for refunds.  $\phi_2$  is the weight given to the class Topic<sub>2</sub> i.e., waiting for an update or reward.  $\phi_3$  is the weight given to the class Topic<sub>3</sub> i.e., reporting or taking legit actions.  $\phi_4$  is the weight given to the class Topic<sub>4</sub> i.e., product never received.  $\phi_5$  is the weight given to the class Topic<sub>5</sub> i.e., showing anger or disappointment. And  $\phi_6$  is the weight given to the class Topic<sub>6</sub> i.e., no communication or confused.

The objective function of PSO aims to reduce the weights of topic classes of Topic<sub>0</sub>, Topic<sub>1</sub>, Topic<sub>2</sub>, Topic<sub>3</sub>, Topic<sub>4</sub>, Topic<sub>5</sub>, and Topic<sub>6</sub>. It also maximizes the weights of classes Topic<sub>7</sub>, Topic<sub>9</sub>, Topic<sub>10</sub>, and Topic<sub>11</sub> with user preferences. Therefore, we can summarize our objective function in Equation 28 below:

$$weight = Max(weight_1) + Min(weight_2) \quad (28)$$

**Table 15. Classification of project's credibility**

<b>Project Credibility</b>	<b>Examples</b>	<b>Topic Classes as representative class</b>
Extremely Low	<p>A project can fall in this category in the following circumstances:</p> <ul style="list-style-type: none"> <li>- If the investors are ready to file a lawsuit against the platform or the content creator.</li> <li>- If more than a year has passed and people has still got nothing</li> <li>- If the emotions of anger and frustrations are getting peak towards the creator of the project.</li> </ul>	Topic_3, Topic_4, Topic_5

<p style="text-align: center;">Low</p>	<p>A project can fall in this category in the following circumstances:</p> <ul style="list-style-type: none"> <li>- If the rewards promised by the creator are still awaiting</li> <li>- None of the investors has received the refunds they claimed for.</li> <li>- If there is a communication lack from the creator side showing that no content such as an update or comment is published after successfully raising the funds.</li> </ul>	<p>Topic_0, Topic_1, Topic_2, Topic_6</p>
<p style="text-align: center;">Normal</p>	<p>A project can fall in this category in the following circumstances:</p> <ul style="list-style-type: none"> <li>- If there are no harsh or hard emotions for the project or its creator by the investors</li> </ul>	<p>Topic_8</p>



	<ul style="list-style-type: none"> <li>- If people are waiting for the product patiently and with positive or neutral emotions.</li> <li>- If some investors have already received something such as a refund or a reward.</li> </ul>	
High	<p>A project can fall in this category in the following circumstances:</p> <ul style="list-style-type: none"> <li>- If the investors are showing contentment and happy emotions for the product.</li> <li>- If the hopes of the investors are very high.</li> <li>- Investors are receiving updates at a continuous rate.</li> </ul>	Topic_7, Topic_11
Extremely High	<p>A project can fall in this category in the following circumstances:</p> <ul style="list-style-type: none"> <li>- If the investors have received the product.</li> </ul>	Topic_9, Topic_10

	- If the investors are talking about the working conditions of the product or describing its features.	
--	--	--

#### 4.4.1 Optimization

We have used PSO algorithm for optimization purpose. PSO is a population-based optimization algorithm. Due to the process of continuous optimization to find the best solution, PSO has become very popular. It comprises of basics features such as the number of particles, and respective velocities of each particle. Each particle upholds two values as particle position and the velocity calculated as given in Equation 9 and 10:

$$V_{particle}(t+1) = W * V_{particle}(t) + coeff1 * var1 * [l_{best}(t) - P_{particle}(t)] + coeff2 * var2 * [g_{best}(t) - P_{particle}(t)] \quad (29)$$

$$P_{particle}(t+1) = P_{particle}(t) + V_{particle}(t+1) \quad (30)$$

Table 16 presents the PSO parameters description.

**Table 16. Parameters of PSO equation**

Equation's Components	Description
$V_{particle}(t)$	Velocity of a Particles at time $t$
$P_{particle}(t)$	Position of a Particle at time $t$
$l_{best}(t)$	Individual or local best solution of a Particle at time $t$

$g_{best}(t)$	The global best solution of a Particle at time $t$
W	Coefficient of inertia Range -> [0.8-1.2]
coeff1	Cognitive coefficient
coeff2	Social coefficient
var1 var2	Random variables

## 4.5 Experimental Setup

In this section, we present the details of the experimental setup. The processes of selecting the resources, preparing data, running the implemented model, training and testing are explained here. This section also covers the analysis of the results of our hybrid approach for predicting topic class of a comment or review. A recommendation mechanism is built upon these predictions that is also capable of optimizations. It accommodates the preferences of users and can recommend projects with maximum credibility levels to them accordingly.

The proceeding subsections shed light on the training and testing phases and elaborate on the data used in each phase. This section concludes with a running example scenario on a sample data of few comments.

Following Table 17 presents the system specifications.

**Table 17. Experimental Setup**

<b>System Details</b>	<b>Characteristics</b>
OS (Operating System)	Ubuntu 18.04.1 LTS

System Memory	32Gb
Primary Programming Language	Python
GPU	Nvidia GForce 1080
Python Version	3.6.1
API	Tensorflow
API Version	1.13

### 4.5.1 Training

In the training cycle we consider the complete dataset and consider the only projects which have at least 10 comments. We have 300 total comments and out of these 300 we use seventy percent (210) for training. From each project following details are extracted.

- 1) Social Media Links
- 2) Profile Picture
- 3) Project Description (Language)
- 4) Current comments

We use a pretrained LDA (trained on bag of words vocabulary already) in this scenario to generate the word and topic probability distributions. These along with the comment which is already preprocessed is used as input for LSTM. The LSTM predicts the categories and we use the ‘sparse cross entropy loss’ of comment category to train the LSTM over the training data. The output of LSTM is then utilized to generate the probability distribution of positive, negative, and neutral comments. This output is then utilized in the authenticity equation to generate the project

recommendation. This process is iterated for multiple hundred epochs till we reached 90 % accuracy on comment classification.

### **4.5.2 Testing**

In the testing phase the same process is followed as training. Any new comment is passed through the preprocessing unit and then its topic and word probability distributions are generated. These topic and word distributions are then fed to the LSTM along with the preprocessed comment into the LSTM. The LSTM in captures the temporal dependencies and predicts the topic distribution for that comment. The only difference is that this prediction is treated as ground truth and then passed at project level where its stored along with other comments of this project. The latest comment along with the previous comments are then used to generate positive, negative and neutral probabilities. These probabilities are then utilized by authentication module to produce the complete recommendation of the project.

## 4.6 Example Scenario

In this section, we present an example to simplify the process. We have taken three random comments and performed the complete process for better understanding. At the first step comments are passed as it is to the hybrid model as shown in Figure 24. The modules in color show where we are focusing currently or in other words are the active modules or tasks.

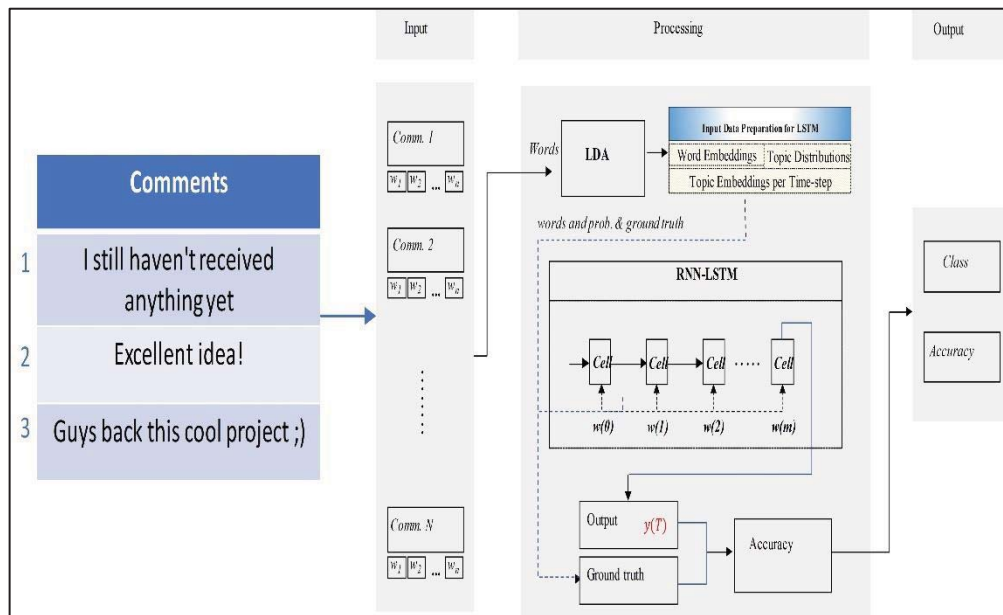


Figure 24. Example scenario input to model – Step 1

At the second step comments are preprocessed. Here, data is being cleansed. Stop words are removed and punctuations are eliminated as shown in Figure 25. The modules in color show where we are focusing currently or in other words are the active modules or tasks. The modules in color gray represents the executed tasks.

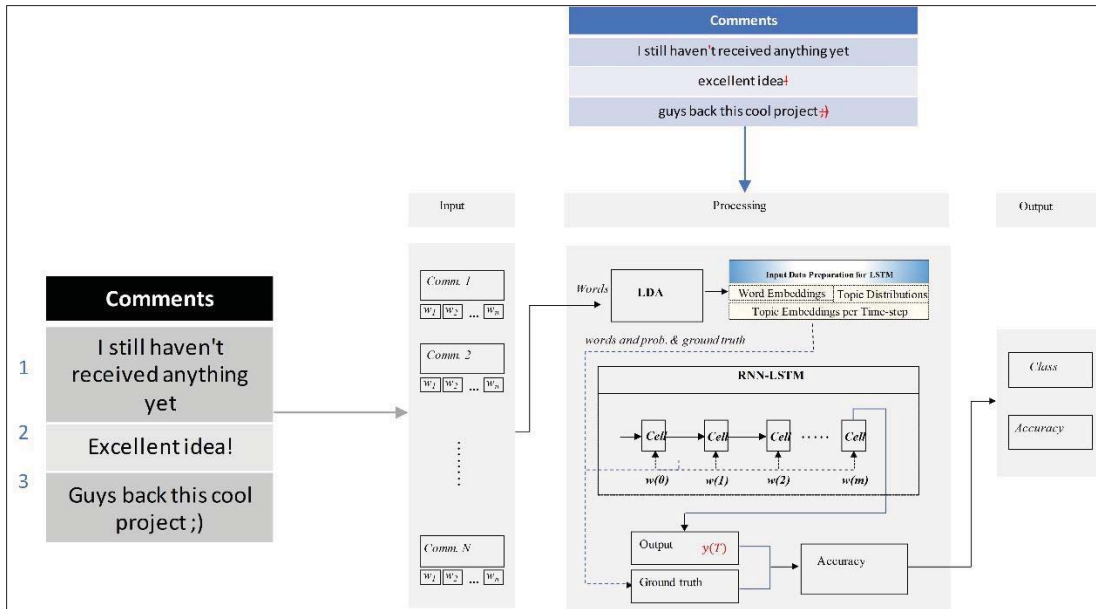


Figure 25. Example scenario input to model – Step 2

After the preprocessing, comments are passed to LDA, where topic probability distributions are generated as shown in the Figure 26.

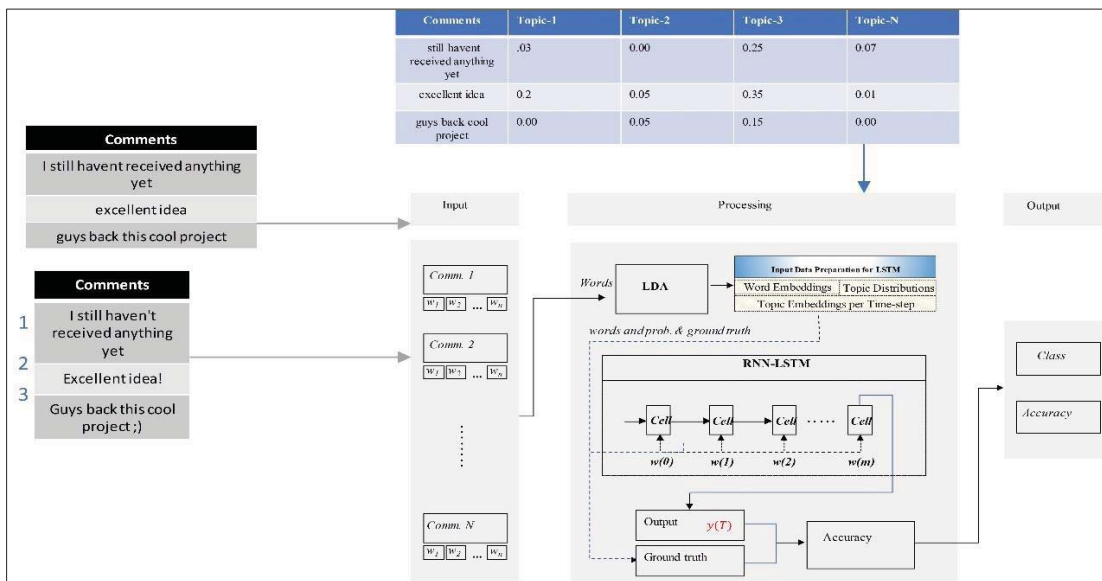


Figure 26. Example scenario input to model – Step 3

After this, topic distribution, temporal data e.g., the published time of the comments are passed to LSTM module as shown in the Figure 27.

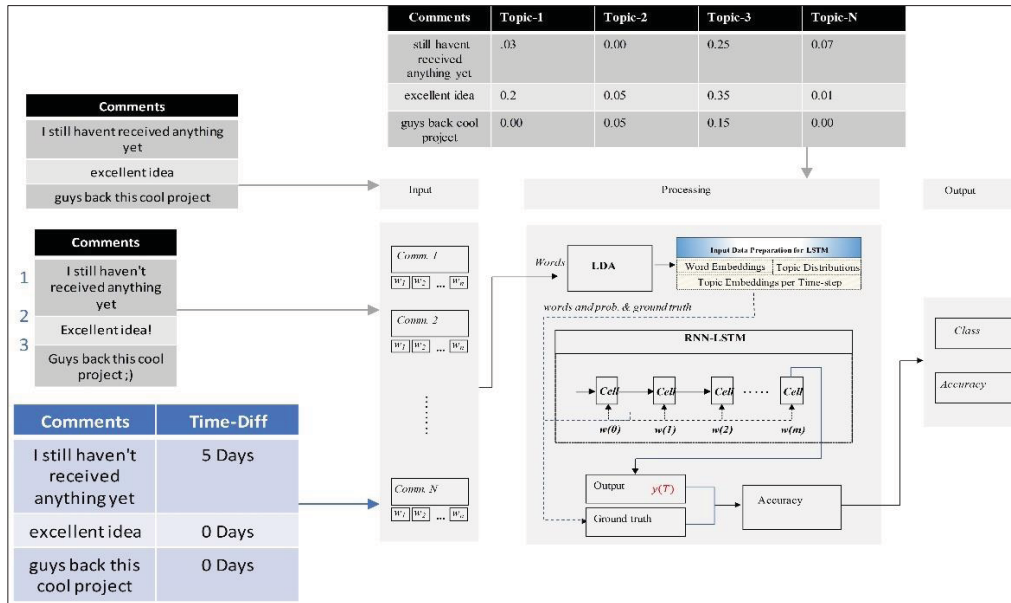


Figure 27. Example scenario input to model – Step 4

LSTM then generates the topic class for each comment as shown in the Figure 28.

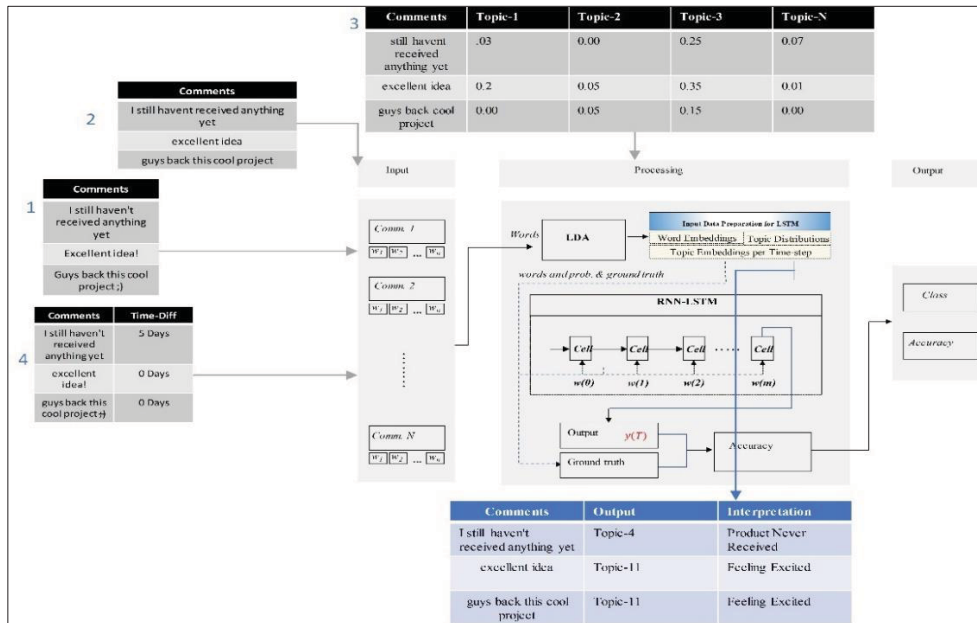


Figure 28. Example scenario input to model – Step 5



Now these topic labels against each comment are stored as ground truth data as shown in the Figure 29.

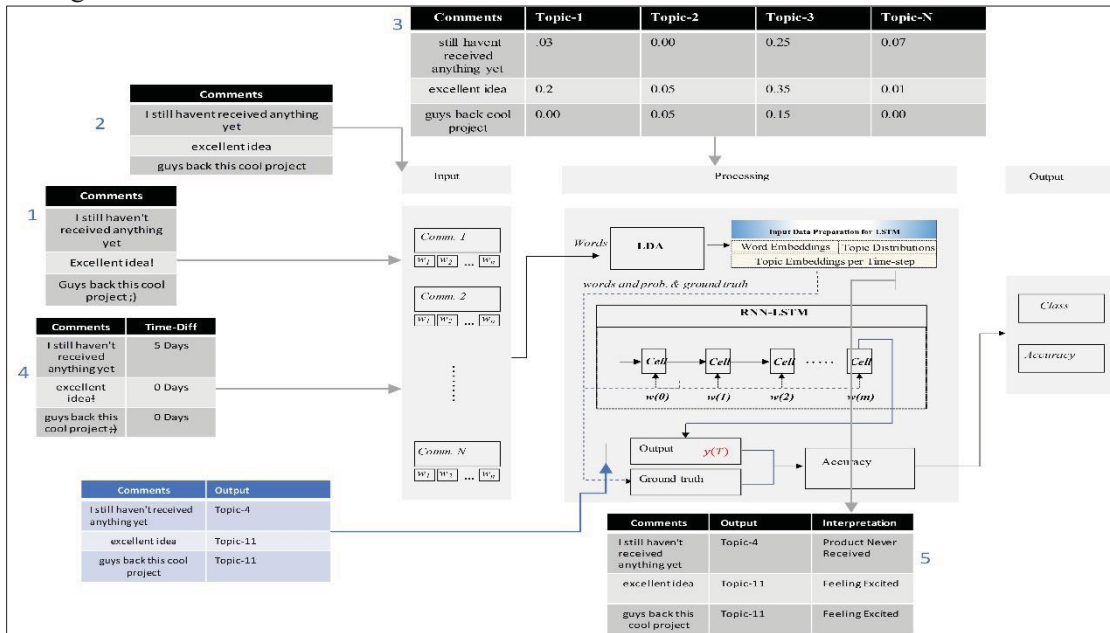


Figure 29. Example scenario input to model – Step 6

After training the model accuracy is estimated as shown in the Figure 30.

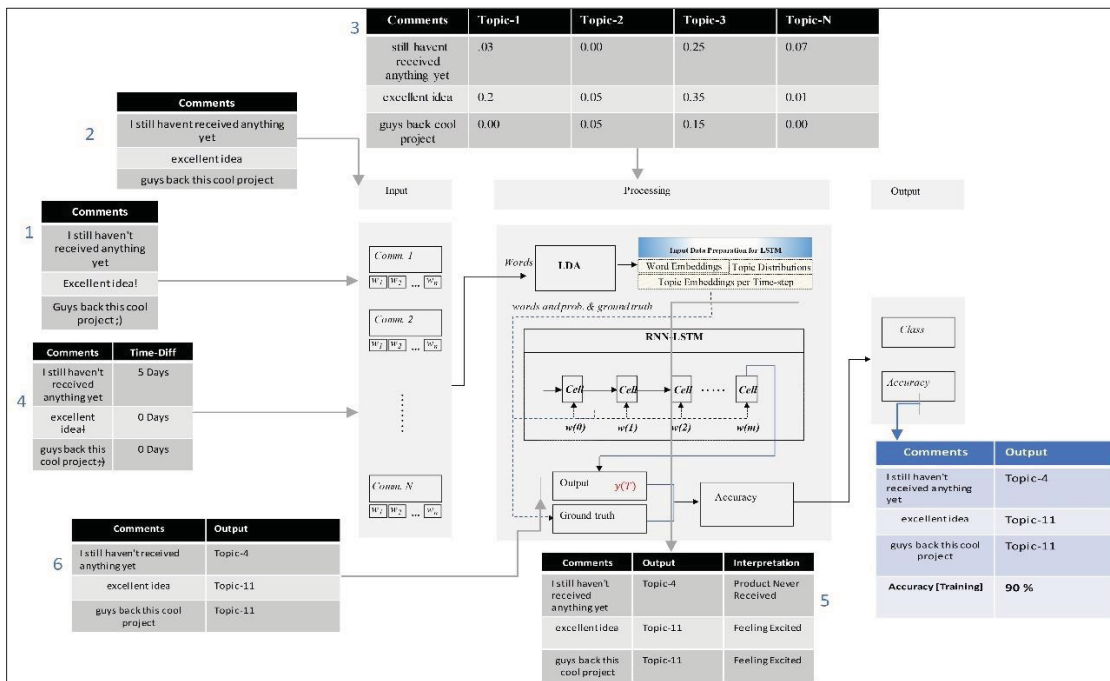


Figure 30. Example scenario input to model – Step 7

Now the overall classification is performed based on percentage of comments falling into each category.

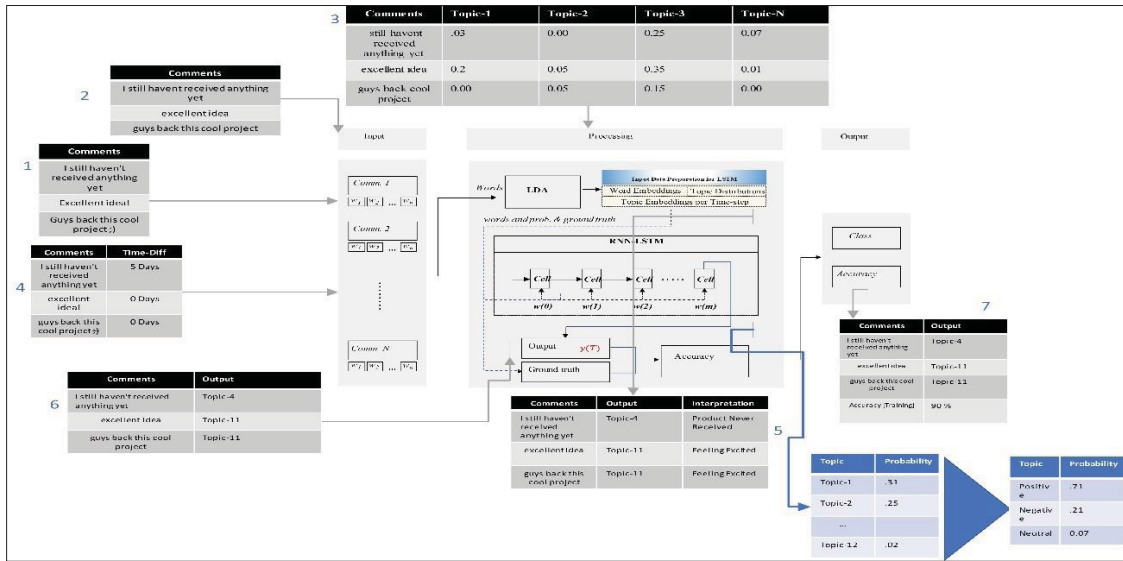


Figure 31. Example scenario input to model – Step 8

At last step, the final project is recommended based on the credibility estimations and optimizations.

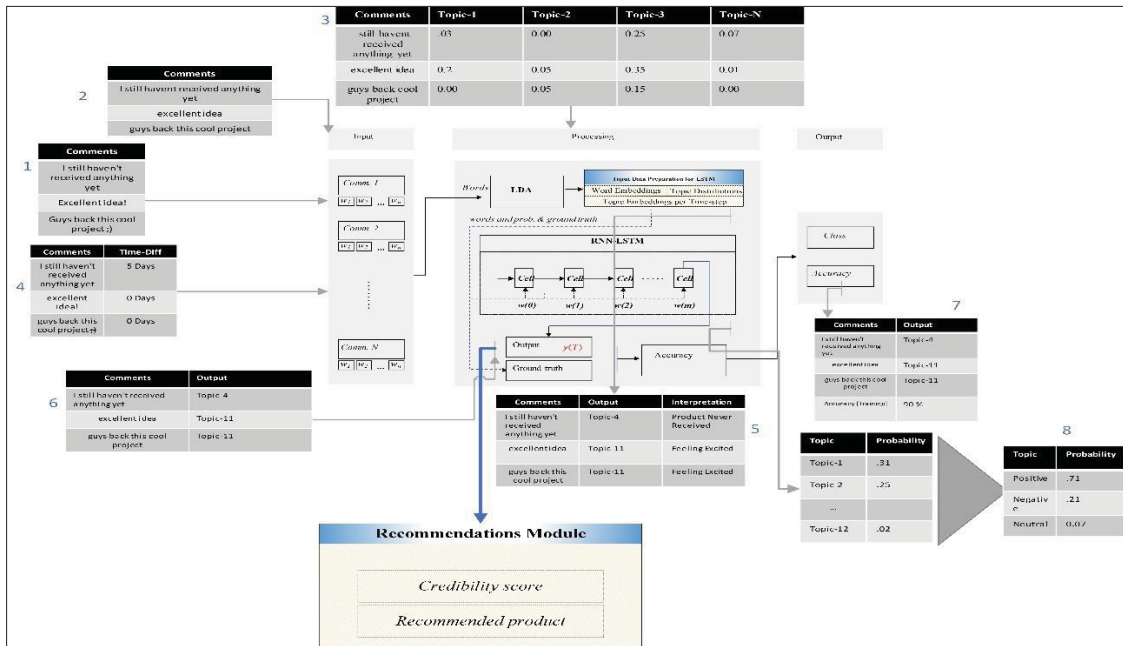


Figure 32. Example scenario input to model – Step 9

# Chapter 5: Experiments and Performance Analysis

In this chapter, all sections are dedicated and composed of a detailed discussion of the performed experiments and their corresponding results. For the sake of simplicity, this section is divided into four subsections. In this section we briefly cover all the results and their investigation.

## 5.1 Optimized Recommendations and Prediction Accuracies

This section presents the primary topic class prediction accuracy of our model as compared with other algorithms referred as simple neural network (NN), and an integrated NN-LDA model. Here we have evaluated the total prediction accuracy of the overall system.

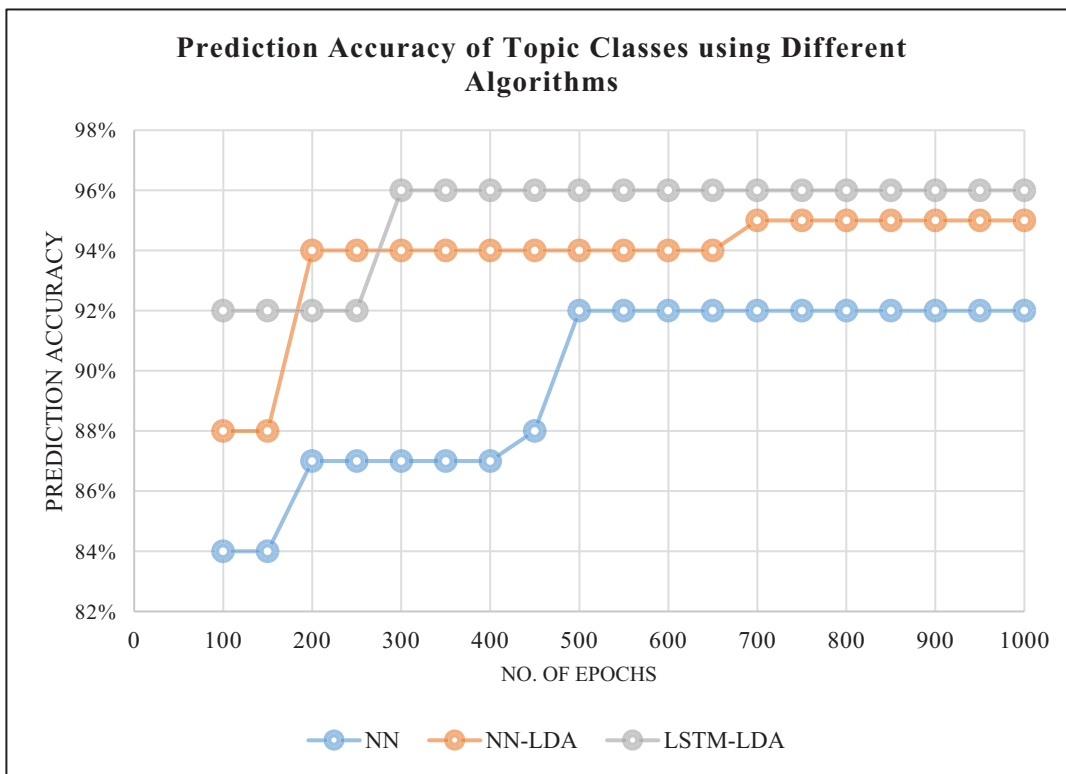


Figure 33. Prediction accuracy of topic classes vs. number of epochs

## 5.2 Prediction Accuracy of Topic Classes

In Figure 34, the prediction accuracy of different algorithms is shown. The proposed LSTM-LDA model is compared with basic NN, and with an LDA based NN model. The results depict that LSTM-LDA is performing better with 96% of prediction accuracy as compared with NN that has approximately 92% of prediction accuracy and NN-LDA that has about 95% of prediction accuracy. However, the performance results of NN-LDA and LSTM-LDA are quite close to each other, the patterns of LSTM-LDA are more stable as compared with NN-LDA.

## 5.3 Prediction Accuracy of Topic Classes for Variable Number of Topics

In this section, we performed an experiment to figure out the optimal number of topics that are well representative of the complete data by experimenting with different number of topics for LDA.

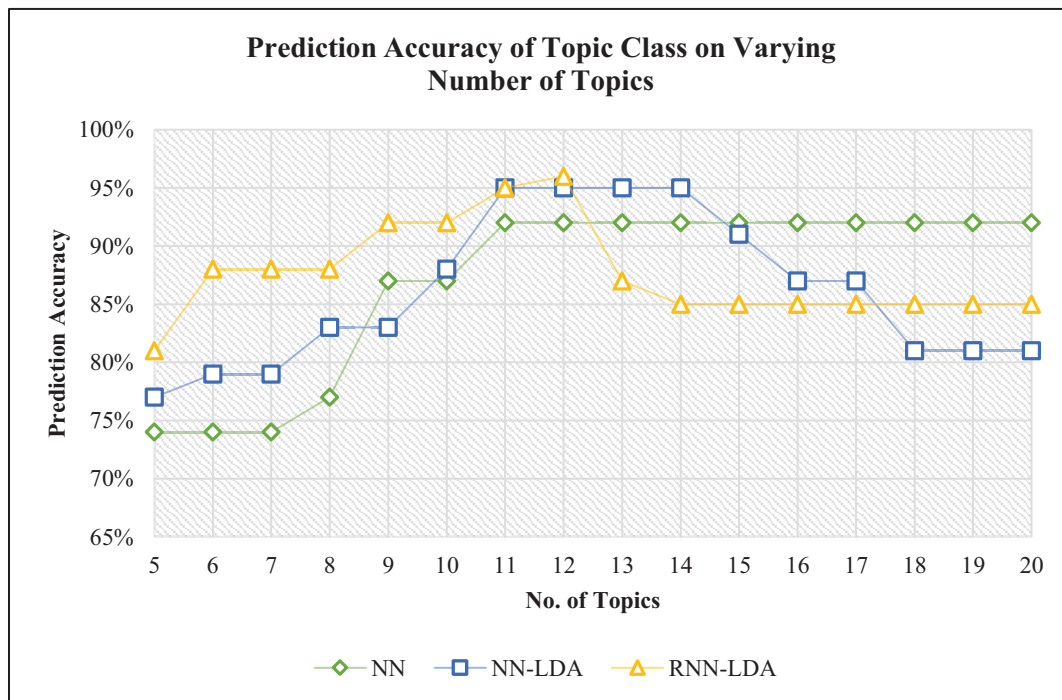


Figure 34. Topic classes and their respective prediction accuracy for variable no. of topics

The number of topics were varied between 5 to 20. With the number of topics changing, the accuracy of prediction also changes as shown in Figure 33. The experiment is performed for all the baseline algorithms under same conditions. The optimal number of topics for all the algorithms lies between the range [10-15]. The maximum prediction accuracy for each algorithm is achieved when the number of topics becomes 12.

## 5.4 Discussion Trends in Suspicious Campaigns

In this section, we performed the experiments to extract the communication trends in crowdfunding comments. For this particular experimentation we chose the category of suspicious campaigns as these campaigns are most likely to have low or extremely low credibility level.

To take a look at the tendencies through the years, we chose distinctive stages of a marketing campaign. The complete lifetime of a campaign is divided into four stages. The first stage is the funding period phase of the project that is usually 40-65 days long. The second stage is after funding period the time between the day funding phase expires and the day of expected delivery. The third stage is one-month period after the expected delivery. The last stage refers to the time after one or more than one year of the expected delivery phase.

Figure 34 shows that the excitement level of investors is quite high during 1<sup>st</sup> stage of the project. They are found to publish good and happy stuff that reveals the emotional level that they currently possess. The level of excitement and happiness seems to drop after the funding period. It decreases from 45% to 27%. The trend and behavior can be explained as after the funding period project enters into implementation phase and the project creators reduce their communication and involvement level with the backers. This communication can be any of any form such as a comment or a regular update from the creator's side. Hence, it starts developing emotions other than excitement as well in the investors.

This excitement might turn into emotions of anger, frustration or disappointment if the delivery date of the project has passed and it still did not get delivered to the investors. On top of this the creator is not communicating properly. Therefore, the comments during this phase mostly are related to refund requests, or rewards claims or include anger emotions.

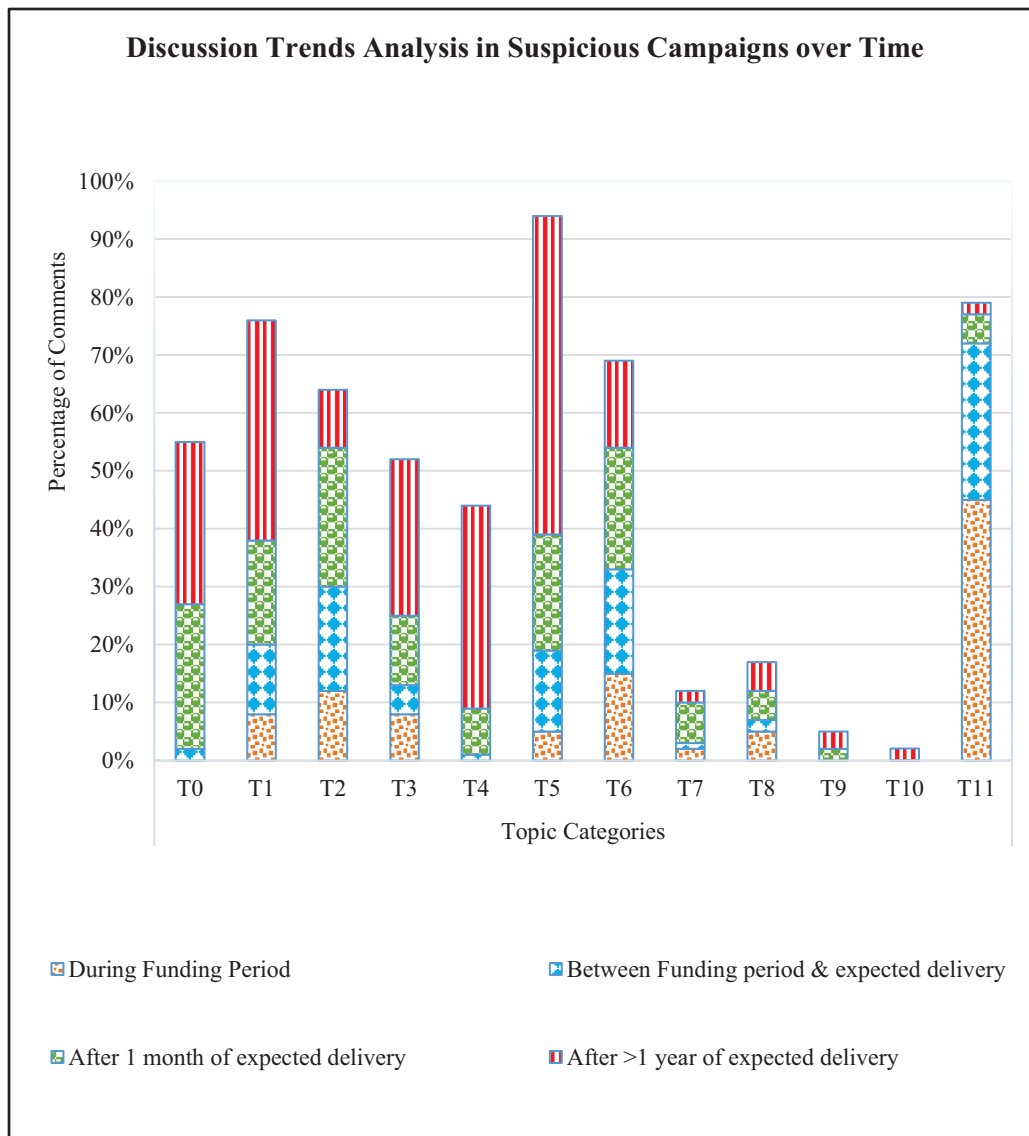


Figure 35. Analysis of Topic classes with time in suspicious campaigns

## 5.5 Analysis of Recommendation Results

To examine the results of our recommendation module, we first selected some ground truth data. We chose 40 projects, 15 non-scams, 10 suspended projects, 8 canceled projects, and 7 successfully funded projects.

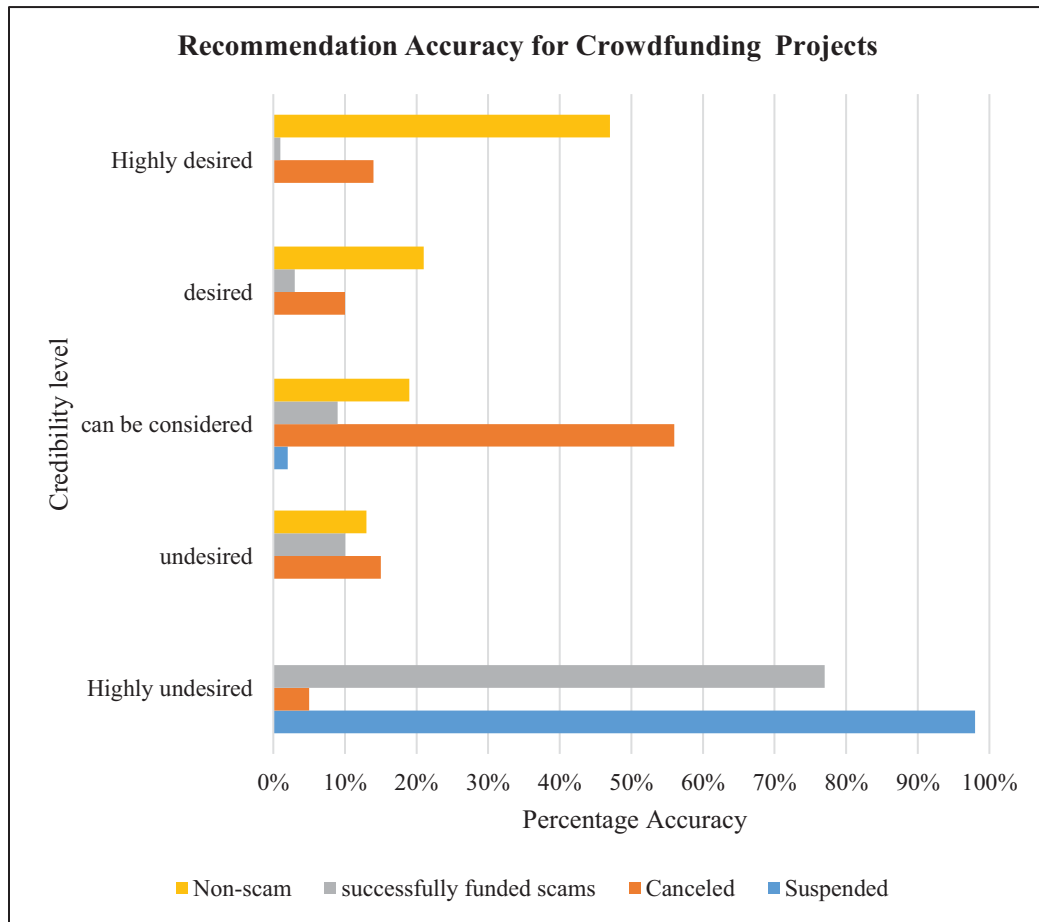


Figure 36. Percentage accuracy of recommended results on ground truth data

These projects data set represents all possible use case scenario, and we can evaluate how well our recommendation system perform on each type of project.

In Figure 37, we have evaluated the computed authenticity levels for the suspended projects. It is clear that the authenticity level for many of the suspended projects is falling in the range [0-0.3] which is true because these projects are being suspended for some suspicious activities.

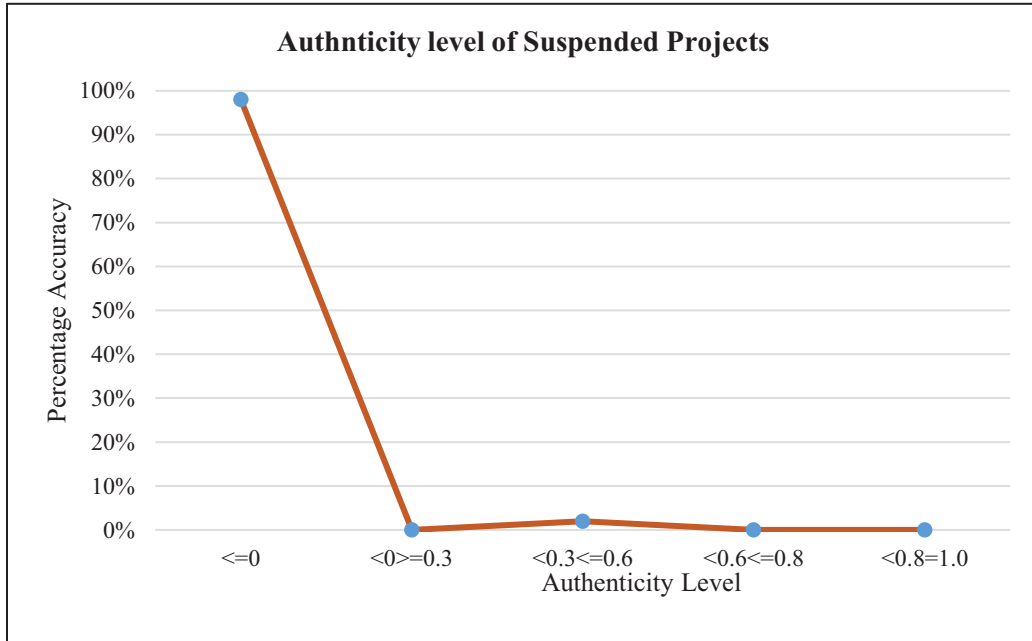


Figure 37. Percentage accuracy of authenticity level of suspended projects

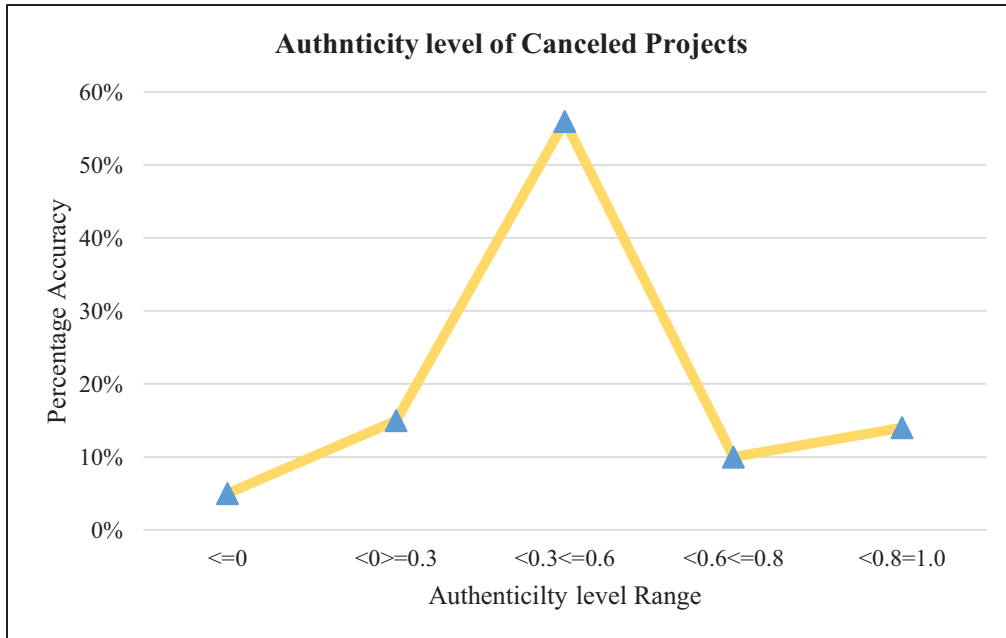
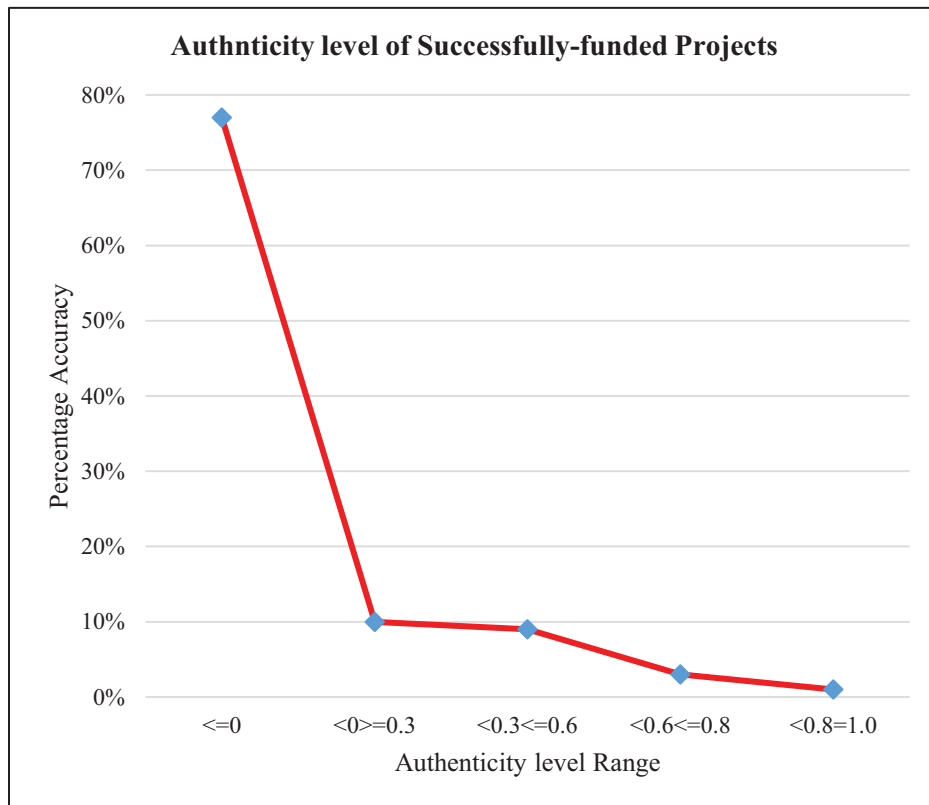


Figure 38. Percentage accuracy of authenticity level of canceled projects



In Figure 38, we have evaluated the computed authenticity levels for the canceled projects. It is clear that the authenticity level for many of the suspended projects is falling in the range [0.3-0.6] which means that these projects can be considered for the investments but keeping the risk factor in mind. These projects are canceled due to multiple reasons such as lack of funding, budget issues, development phases etc.



**Figure 39. Percentage accuracy of authenticity level of successfully-funded projects**

Figure 39 presents the recommendation results accuracy on the successfully funded yet scam projects. The results show that in most cases the authenticity level range is between 0 to 0.4. It shows that though these projects got successfully funded but people are not happy with the

development progress and many elements when combined together happen to increase the risk involved.

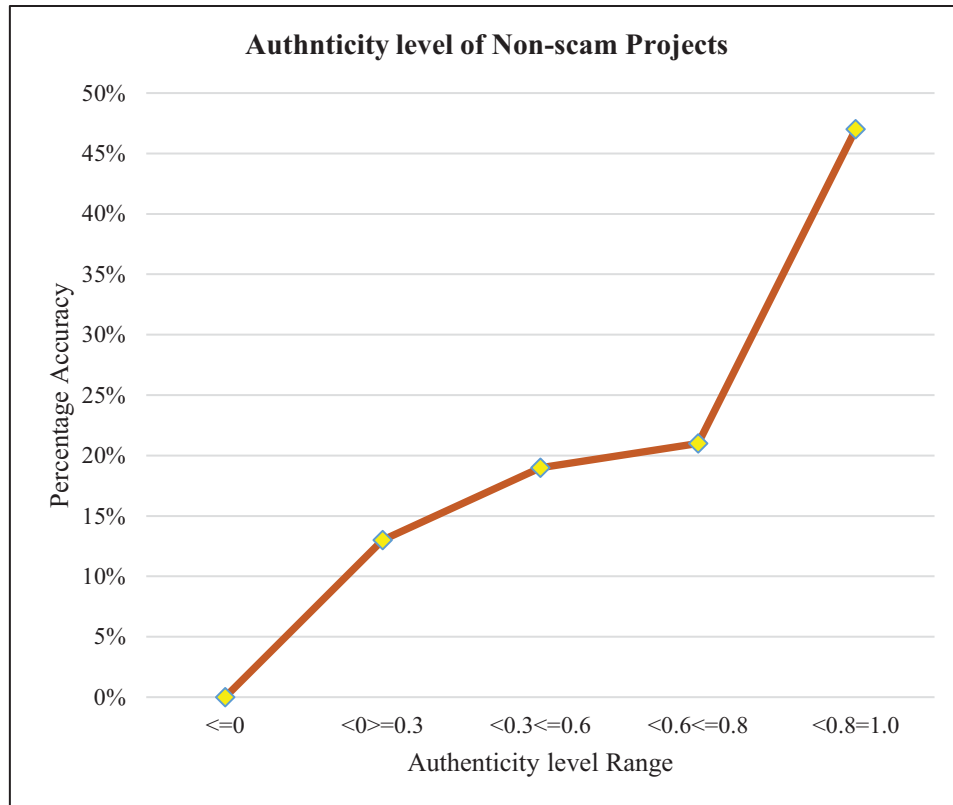


Figure 40. Percentage accuracy of authenticity level of non-scam projects

Figure 40, shows an interesting trend. It presents the authenticity level estimation accuracy for non-scam projects. The trend shows that for non-scam projects, it is less likely for a project to have factors which can result into lower levels of authenticity range. Mostly projects are falling in the range of 0.4 to 0.9, showing that comments are falling in the gray range which usually is for less risky projects.

Following Table 18 presents the decision boundaries and ranges used by the recommendation module.

**Table 18: Parameter settings for the optimization algorithm**

<b>Decision</b>	<b>Credibility Level of a Project</b>	<b>Example Topic Classes</b>	<b>Range of Authenticity (Score)</b>
Highly undesired	Extremely Low	Topic_3, Topic_4, Topic_5	[> 0 <=0.3]
Not desired	Low	Topic_0, Topic_1, Topic_2, Topic_6	[> 0.3 <=0.6]
Can be considered	Normal	Topic_8	[> 0.6 <=0.7]
Desired	High	Topic_7, Topic_11	[> 0.8 <= 0.9]
Highly Desired	Extremely High	Topic_9, Topic_10	[>0.9 - <= 1.0]

To check the testing error of the proposed model, different learning rates are used i.e., 0.1, 0.01 and 0.001 referred as LR\_0.1, LR\_0.01, and LR\_0.001 in Figure 41. The method used to calculate the testing error is Root Mean Square Error (RSME). It can be observed that the testing errors start to get decreased if the learning rate gets smaller. For example, for LR\_0.001, the error

becomes less than 0.2. It represents that with a smaller value of learning rate, the performance of the system gets better.

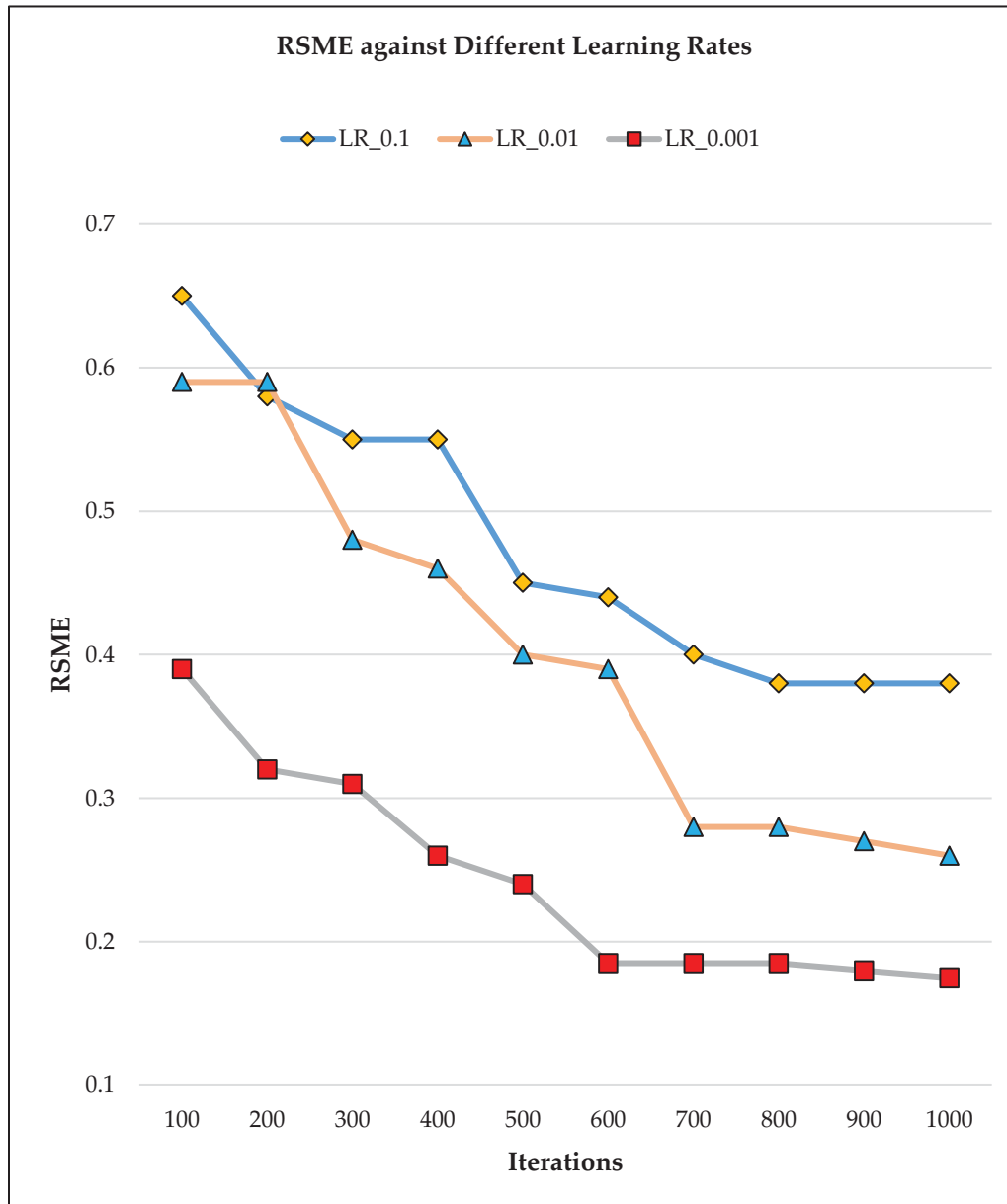


Figure 41. Testing error against different learning rates

## Chapter 6: Conclusions

In this this, we have proposed an optimized recommendation mechanism to recommend a most reliable item to the user. The proposed system uses textual and non-textual data. This system is developed to help the users in selecting reliable and trustworthy options in their preferred categories. The proposed system is built on a hybrid model of RNN-LSTM and topic modeling that joins the benefits of both (1) LSTMs that captures time dependencies for class and topic prediction, and (2) topic modeling that extracts topics that well summarize the content. A case study on crowdfunding is performed to analyze and test the proposed system's behavior. We have also embedded an optimized recommendation strategy based on a project's credibility.

Our results have complemented the existing studies in showing that the hybrid methods perform well and better in topic identification. This joint model of LSTM-LDA, that utilizes words and topic embeddings along with the temporal data achieves 96% of accuracy in accurately predicting the topic categories. The topics classes discovered were also evaluated in the context of helping investors identify the suspicious campaigns. The prediction quality can be improved if we discover different configurations of comments with respect to the timeline of a project. We experimented this by dividing the comments into five various batches of comments, each batch belongs to one specific timeline and consists of around 50 to 70 comments. We have not considered projects that have less than 50 comments to maintain the quality of the results.

In Table 19, the results for this experiment are summarized. It can be observed that the classification results are getting accurate towards the batches that have most recent comments. For example, Comments\_batch 5 has the highest accuracy of having most definitive topic classes towards scam or non-scams as it has the newest comments so far. If we elaborate it more, we can

state that there are less chances that in the latest comments people are talking about multiple topics, they most probably are sharing the same thoughts.

**Table 19: Classification accuracy of projects based on different batch sizes of comments**

No.	Time Period	Classification Accuracy
Comments_Batch 1	Comments after the campaign launched till the fundraising period	20%
Comments_Batch 2	Comments after the funding phase	55%
Comments_Batch 3	Comments before the expected delivery date	67%
Comments_Batch 4	Comments after the expected delivery date	83%
Comments_Batch 5	Top 60 new comments	88%

The data available on internet is advantageous for many applications and organizations. However, the appropriate and effective usage and analysis becomes a hefty challenge primarily for the recommendation systems. As recommendation systems need continuous and timely upgradation with respect to user preferences, a continuous struggle is needed to improve the system. User-generated content has been quite popular in recent era as many researchers have used

content such as blogs, comments and reviews, for their experiments in different domain-specific applications. Nevertheless, similar type of content in crowdfunding is often paid no attention. For example, the comments of backers still need attention and in-depth analysis. The risks and challenges of suspicious or fraudulent actions on such forums can be reduced by paying more attention towards apt and timely analysis of the content produced by the participants.

In summary, there are many developed applications for recommendation systems in different fields. Our proposed approach is a novel approach to recommend a credible item to best of our knowledge. Moreover, none of the works have focused on crowdfunding comments to find discussion trends and their impact towards project credibility. Hence, in crowdfunding, this approach can be used to recommend safe or secure projects to investors.

The objective of this study is to overcome the limitations of topic models and deep learning and get the most out of both approaches. The main objectives include:

- 3) Finding ways to preserve the contextual dependencies as traditional topic models are based on bag-of words approach, so there is high probability to miss the contextual and temporal dependencies.
- 4) Recommendation tools are in use since a long time now, finding credibility of the recommended product or location is a potential target of this research.

We summarize the contributions of this thesis as follows: (1) a basic and hybrid method is proposed for reliable and trustworthy recommendations. This method is using dynamic user preferences and word embeddings that is capable of modeling user preferences and words representations in a joint and dynamic manner. In the context of document streams this modeling is done in the same semantic space. This enables the effective measurement of the semantic similarity between the user's preferences and the words. (2) The proposed algorithm is to deduce the dynamic embeddings of both documents and words. Our optimization module works based on

an objective function. We propose a credibility measurement approach for secure recommendation

(4) The effectiveness of the various embeddings-based model is verified along with the optimization algorithm and the objective function, on crowdfunding projects and tourism blogs. We compared our proposed model with baseline algorithms as NN-LDA, NN, and SVM-LDA etc. The results show that our proposed method outperforms similar state-of-the-art methods significantly.



# References

- 1) Liu, L., Tang, L., Dong, W., Yao, S. and Zhou, W., 2016. An overview of topic modeling and its current applications in bioinformatics. SpringerPlus, 5(1), p.1608.
- 2) Blei, D., Carin, L. and Dunson, D., 2010. Probabilistic Topic Models: A focus on graphical model design and applications to document and image analysis. IEEE signal processing magazine, 27(6), p.55.
- 3) Wang, C. and Blei, D.M., 2011, August. Collaborative topic modeling for recommending scientific articles. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 448-456). ACM.
- 4) Hu, Bo, and Martin Ester. "Spatial topic modeling in online social media for location recommendation." In Proceedings of the 7th ACM conference on Recommender systems, pp. 25-32. ACM, 2013.
- 5) Hariri, N., Mobasher, B. and Burke, R., 2012, September. Context-aware music recommendation based on latent topic sequential patterns. In Proceedings of the sixth ACM conference on Recommender systems (pp. 131-138). ACM.
- 6) She, J. and Chen, L., 2014, April. Tomoha: Topic model-based hashtag recommendation on twitter. In Proceedings of the 23rd International Conference on World Wide Web (pp. 371-372). ACM.
- 7) Korfiatis, N., Stamolampros, P., Kourouthanassis, P. and Sagiadinos, V., 2019. Measuring service quality from unstructured data: A topic modeling application on airline passengers' online reviews. Expert Systems with Applications, 116, pp.472-486.
- 8) Linden, G., Smith, B. and York, J., 2003. Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet computing, (1), pp.76-80.
- 9) Jin, M., Luo, X., Zhu, H. and Zhuo, H.H., 2018, June. Combining deep learning and topic modeling for review understanding in context-aware recommendation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (pp. 1605-1614).
- 10) Pergola, G., Gui, L. and He, Y., 2019. TDAM: A topic-dependent attention model for sentiment analysis. Information Processing & Management, 56(6), p.102084..
- 11) Karami, A., 2015. Fuzzy topic modeling for medical corpora. University of Maryland, Baltimore County.
- 12) Asuncion, H.U., Asuncion, A.U. and Taylor, R.N., 2010, May. Software traceability with topic modeling. In 2010 ACM/IEEE 32nd International Conference on Software Engineering (Vol. 1, pp. 95-104). IEEE.
- 13) Ghosh, D. and Guha, R., 2013. What are we 'tweeting' about obesity? Mapping tweets with topic modeling and Geographic Information System. Cartography and geographic information science, 40(2), pp.90-102.
- 14) DiMaggio, P., Nag, M. and Blei, D., 2013. Exploiting affinities between topic modeling and the sociological perspective on culture: Application to newspaper coverage of US government arts funding. Poetics, 41(6), pp.570-606.
- 15) Brants, T., Chen, F. and Tsochantaridis, I., 2002, November. Topic-based document segmentation with probabilistic latent semantic analysis. In Proceedings of the eleventh international conference on Information and knowledge management (pp. 211-218). ACM.
- 16) Brants, T., Chen, F. and Tsochantaridis, I., 2002, November. Topic-based document

- segmentation with probabilistic latent semantic analysis. In Proceedings of the eleventh international conference on Information and knowledge management (pp. 211-218). ACM.
- 17) Blei, D.M., Ng, A.Y., and Jordan, M.I., —Latent Dirichlet Allocation, Journal of Machine Learning Research, 3, 2003, 993-1022.
  - 18) Blei, D. and Lafferty, J., 2006. Correlated topic models. Advances in neural information processing systems, 18, p.147.
  - 19) Dumais, S.T., 2004. Latent semantic analysis. Annual review of information science and technology, 38(1), pp.188-230.
  - 20) Landauer, T.K., Foltz, P.W. and Laham, D., 1998. An introduction to latent semantic analysis. Discourse processes, 25(2-3), pp.259-284.
  - 21) Landauer, T.K., 2002. Applications of latent semantic analysis. In Proceedings of the Annual Meeting of the Cognitive Science Society (Vol. 24, No. 24).
  - 22) Sidorov, G., 2019. Latent Semantic Analysis (LSA): Reduction of Dimensions. In Syntactic n-grams in Computational Linguistics (pp. 17-19). Springer, Cham.
  - 23) Sehra, S., Singh, J. and Rai, H., 2017. Using latent semantic analysis to identify research trends in openstreetmap. ISPRS International Journal of Geo-Information, 6(7), p.195.
  - 24) Hofmann, T., 1999, July. Probabilistic latent semantic analysis. In Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence (pp. 289-296). Morgan Kaufmann Publishers Inc..
  - 25) Hofmann, T., —Unsupervised learning by probabilistic latent semantic analysis, Machine Learning, 42 (1), 2001, 177- 196.
  - 26) Kakkonen, T., Myller, N., Sutinen, E., and Timonen, J., —Comparison of Dimension Reduction Methods for Automated Essay Grading, Educational Technology & Society, 11 (3), 2008, 275-288.
  - 27) Liu, S., Xia, C., and Jiang, X., —Efficient Probabilistic Latent Semantic Analysis with Sparsity Control” IEEE International Conference on Data Mining, 2010, 905-910.
  - 28) Romberg, S., Hörster, E., and Lienhart, R., —Multimodal pLSA on visual features and tags, The Institute of Electrical and Electronics Engineers Inc., 2009, 414-417.
  - 29) Wu, H., Wang, Y., and Cheng, X., —Incremental probabilistic latent semantic analysis for automatic question recommendation, ACM New York, NY, USA, 2008, 99-106.
  - 30) Zhi-Yong Shen, Z.Y., Sun, J., and Yi-Dong Shen, Y.D., —Collective Latent Dirichlet Allocation, Eighth IEEE International Conference on Data Mining, pages 1019–1025, 2008.
  - 31) Rosen-Zvi, M., Griffiths, T., Steyvers, M. and Smyth, P., 2004, July. The author-topic model for authors and documents. In Proceedings of the 20th conference on Uncertainty in artificial intelligence (pp. 487-494). AUAI Press.
  - 32) X. Wang and A. McCallum. —Topics over time: a non-markov continuous-time model of topical trends. In International conference on Knowledge discovery and data mining, pages 424–433, 2006.
  - 33) McCallum, A., Wang, X., and Corrada-Emmanuel, A., —Topic and role discovery in social networks with experiments on enron and academic email, Journal of Artificial Intelligence Research, 30 (1), 2007, 249- 272.
  - 34) Bao, S., Xu, S., Zhang, L., Yan, R., Su, Z., Han, D., and Yu, Y., —Joint Emotion-Topic Modeling for Social Affective Text Mining, Data Mining, 2009. ICDM\_09. Ninth IEEE International Conference, 2009, 699-704.
  - 35) Kakkonen, T., Myller, N., and Sutinen, E., —Applying latent Dirichlet allocation to automatic essay grading, Lecture Notes in Computer Science, 4139, 2006, 110-120.
  - 36) Bergholz, A., Chang, J., Paaß, G., Reichartz, F., and Strobel, S., —Improved phishing

- detection using model-based features, 2008. [17] Lee, S., Baker, J., Song, J., and Wetherbe, J.C., —An Empirical Comparison of Four Text Mining Methods, Proceedings of the 43rd Hawaii International Conference on System Sciences, 2010.
- 37) Cheng, X., Yan, X., Lan, Y. and Guo, J., 2014. Btm: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering*, 26(12), pp.2928-2941.
  - 38) Li, C., Wang, H., Zhang, Z., Sun, A. and Ma, Z., 2016, July. Topic modeling for short texts with auxiliary word embeddings. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval* (pp. 165-174). ACM.
  - 39) Lau, R.Y., Xia, Y. and Ye, Y., 2014. A probabilistic generative model for mining cybercriminal networks from online social media. *IEEE Computational intelligence magazine*, 9(1), pp.31-43.
  - 40) Zhai, C., 2008. Statistical language models for information retrieval. *Synthesis Lectures on Human Language Technologies*, 1(1), pp.1-141.
  - 41) Srikanth, M. and Srihari, R., 2002, August. Biterm language models for document retrieval. In *SIGIR* (Vol. 2, pp. 425-426).
  - 42) Torbati, A.H.H.N. and Picone, J., 2015. A doubly hierarchical Dirichlet process hidden Markov model with a non-ergodic structure. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(1), pp.174-184.
  - 43) Edwards, D.S., Globy s Inc, 2010. Targeted customer offers based on predictive analytics. U.S. Patent Application 12/546,449.
  - 44) Wang, S., Chen, Z. and Liu, B., 2016, April. Mining aspect-specific opinion using a holistic lifelong topic model. In *Proceedings of the 25th international conference on world wide web* (pp. 167-176). International World Wide Web Conferences Steering Committee.
  - 45) Zhao, W.X., Jiang, J., Yan, H. and Li, X., 2010, October. Jointly modeling aspects and opinions with a MaxEnt-LDA hybrid. In *Proceedings of the 2010 conference on empirical methods in natural language processing* (pp. 56-65). Association for Computational Linguistics.
  - 46) Mukherjee, A. and Liu, B., 2012, July. Aspect extraction through semi-supervised modeling. In *Proceedings of the 50th annual meeting of the association for computational linguistics: Long papers-volume 1* (pp. 339-348). Association for Computational Linguistics.
  - 47) Poria, S., Cambria, E. and Gelbukh, A., 2016. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, 108, pp.42-49.
  - 48) Chong, W., Blei, D. and Li, F.F., 2009, June. Simultaneous image classification and annotation. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1903-1910). IEEE.
  - 49) Lienhart, R. and Hauke, R., 2009, June. Filtering adult image content with topic models. In *2009 IEEE International Conference on Multimedia and Expo* (pp. 1472-1475). IEEE.
  - 50) Su, Y. and Jurie, F., 2012. Improving image classification using semantic attributes. *International journal of computer vision*, 100(1), pp.59-77.
  - 51) Linstead, E., Rigor, P., Bajracharya, S., Lopes, C. and Baldi, P., 2007, November. Mining concepts from code with probabilistic topic models. In *Proceedings of the twenty-second IEEE/ACM international conference on Automated software engineering* (pp. 461-464). ACM.
  - 52) Sun, X., Liu, X., Li, B., Duan, Y., Yang, H. and Hu, J., 2016, May. Exploring topic models in software engineering data analysis: A survey. In *2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and*

- Parallel/Distributed Computing (SNPD) (pp. 357-362). IEEE.
- 53) Saeidi, A.M., Hage, J., Khadka, R. and Jansen, S., 2015, May. ITMViz: interactive topic modeling for source code analysis. In Proceedings of the 2015 IEEE 23rd International Conference on Program Comprehension (pp. 295-298). IEEE Press.
  - 54) Smith, A., Kumar, V., Boyd-Graber, J., Seppi, K. and Findlater, L., 2018, March. Closing the loop: User-centered design and evaluation of a human-in-the-loop topic modeling system. In 23rd International Conference on Intelligent User Interfaces (pp. 293-304). ACM.
  - 55) Charoenwet, W., 2018, October. A Digital Collection Study and Framework Exploration—Applying Textual Analysis on Source Code Collection. In 2018 3rd Digital Heritage International Congress (DigitalHERITAGE) held jointly with 2018 24th International Conference on Virtual Systems & Multimedia (VSMM 2018) (pp. 1-8). IEEE.
  - 56) Bao, S., Xu, S., Zhang, L., Yan, R., Su, Z., Han, D. and Yu, Y., 2009, December. Joint emotion-topic modeling for social affective text mining. In 2009 Ninth IEEE International Conference on Data Mining (pp. 699-704). IEEE.
  - 57) Rao, Y., 2015. Contextual sentiment topic model for adaptive social emotion classification. *IEEE Intelligent Systems*, 31(1), pp.41-47.
  - 58) Zhu, C., Zhu, H., Ge, Y., Chen, E. and Liu, Q., 2014, December. Tracking the evolution of social emotions: A time-aware topic modeling perspective. In 2014 IEEE International Conference on Data Mining (pp. 697-706). IEEE.
  - 59) Li, X., Rao, Y., Chen, Y., Liu, X. and Huang, H., 2016, March. Social emotion classification via reader perspective weighted model. In Thirtieth AAAI Conference on Artificial Intelligence.
  - 60) Chae, J., Thom, D., Bosch, H., Jang, Y., Maciejewski, R., Ebert, D.S. and Ertl, T., 2012, October. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *IEEE VAST* (pp. 143-152).
  - 61) Keane, N., Yee, C. and Zhou, L., 2015, June. Using topic modeling and similarity thresholds to detect events. In Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation (pp. 34-42).
  - 62) Mishler, A., Crabb, E.S., Paletz, S., Hefright, B. and Golonka, E., 2015, August. Using structural topic modeling to detect events and cluster Twitter users in the Ukrainian crisis. In *International Conference on Human-Computer Interaction* (pp. 639-644). Springer, Cham.
  - 63) Vavliakis, K.N., Symeonidis, A.L. and Mitkas, P.A., 2013. Event identification in web social media through named entity recognition and topic modeling. *Data & Knowledge Engineering*, 88, pp.1-24.
  - 64) Wang, C. and Blei, D.M., 2011, August. Collaborative topic modeling for recommending scientific articles. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 448-456). ACM.
  - 65) Zheng, L., Noroozi, V. and Yu, P.S., 2017, February. Joint deep modeling of users and items using reviews for recommendation. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (pp. 425-434). ACM.
  - 66) Lee, S.S., Chung, T. and McLeod, D., 2011, April. Dynamic item recommendation by topic modeling for social networks. In 2011 Eighth International Conference on Information Technology: New Generations (pp. 884-889). IEEE.
  - 67) Bansal, T., Belanger, D. and McCallum, A., 2016, September. Ask the gru: Multi-task learning for deep text recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems (pp. 107-114). ACM.
  - 68) Tran, V.C., Hwang, D. and Nguyen, N.T., 2018. Hashtag recommendation approach based

- on content and user characteristics. *Cybernetics and Systems*, 49(5-6), pp.368-383.
- 69) Agarwal, D. and Chen, B.C., 2010, February. fLDA: matrix factorization through latent dirichlet allocation. In *Proceedings of the third ACM international conference on Web search and data mining* (pp. 91-100). ACM.
  - 70) Zhu, J., Ahmed, A. and Xing, E.P., 2012. MedLDA: maximum margin supervised topic models. *Journal of Machine Learning Research*, 13(Aug), pp.2237-2278.
  - 71) Terragni, S., Fersini, E. and Messina, E., 2020. Constrained Relational Topic Models. *Information Sciences*, 512, pp.581-594.
  - 72) Tian, K., Revelle, M. and Poshyvanyk, D., 2009, May. Using latent dirichlet allocation for automatic categorization of software. In *2009 6th IEEE International Working Conference on Mining Software Repositories* (pp. 163-166). IEEE.
  - 73) Ramage, D., Hall, D., Nallapati, R. and Manning, C.D., 2009, August. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1* (pp. 248-256). Association for Computational Linguistics.
  - 74) Ding, W., Song, X., Guo, L., Xiong, Z. and Hu, X., 2013, November. A novel hybrid HDP-LDA model for sentiment analysis. In *Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)-Volume 01* (pp. 329-336). IEEE Computer Society.
  - 75) Dos Santos, A.R. and Gonzaga, A., Face Recognition Based on LDA and SOM Neural Nets.
  - 76) Blei, D.M. and Moreno, P.J., 2001, September. Topic segmentation with an aspect hidden Markov model. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 343-348). ACM.
  - 77) Sizov, S., 2012. Latent geospatial semantics of social media. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(4), p.64.
  - 78) Tan, Y. and Ou, Z., 2010. Topic-weak-correlated latent dirichlet allocation. In *2010 7th International Symposium on Chinese Spoken Language Processing* (pp. 224-228). IEEE.
  - 79) Lin, C. and He, Y., 2009, November. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM conference on Information and knowledge management* (pp. 375-384). ACM.
  - 80) Lin, C., He, Y. and Everson, R., 2011, November. Sentence subjectivity detection with weakly-supervised learning. In *Proceedings of 5th International Joint Conference on Natural Language Processing* (pp. 1153-1161).
  - 81) Zhao, W.X., Jiang, J., Weng, J., He, J., Lim, E.P., Yan, H. and Li, X., 2011, April. Comparing twitter and traditional media using topic models. In *European conference on information retrieval* (pp. 338-349). Springer, Berlin, Heidelberg.
  - 82) Zhang, X.P., Zhou, X.Z., Huang, H.K., Feng, Q., Chen, S.B. and Liu, B.Y., 2011. Topic model for chinese medicine diagnosis and prescription regularities analysis: case on diabetes. *Chinese journal of integrative medicine*, 17(4), pp.307-313.
  - 83) Andrzejewski, D.M., Craven, M. and Zhu, X., 2010. Incorporating domain knowledge in latent topic models (Doctoral dissertation, University of Wisconsin--Madison).
  - 84) Das, P., Srihari, R. and Fu, Y., 2011, October. Simultaneous joint and conditional modeling of documents tagged from two perspectives. In *Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 1353-1362). ACM.
  - 85) Wang, H., Ding, Y., Tang, J., Dong, X., He, B., Qiu, J. and Wild, D.J., 2011. Finding complex biological relationships in recent PubMed articles using Bio-LDA. *PloS one*, 6(3), p.e17243.

- 86) Zhai, K., Boyd-Graber, J. and Asadi, N., 2011. Using variational inference and MapReduce to scale topic modeling. arXiv preprint arXiv:1107.3765.
- 87) Mohammadi, M., Raahemi, B., Akbari, A., Nassersharif, B. and Moeinzadeh, H., 2012. Improving linear discriminant analysis with artificial immune system-based evolutionary algorithms. *Information Sciences*, 189, pp.219-232.
- 88) Mao, X.L., Ming, Z.Y., Chua, T.S., Li, S., Yan, H. and Li, X., 2012, July. SSHLDA: a semi-supervised hierarchical topic model. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning* (pp. 800-809). Association for Computational Linguistics.
- 89) Hu, Y., John, A., Wang, F. and Kambhampati, S., 2012, July. Et-lda: Joint topic modeling for aligning events and their twitter feedback. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*.
- 90) Nguyen, H.N., Van Le, T., Le, H.S. and Pham, T.V., 2014, December. Domain specific sentiment dictionary for opinion mining of vietnamese text. In *International Workshop on Multi-disciplinary Trends in Artificial Intelligence* (pp. 136-148). Springer, Cham.
- 91) Yuan, J., Gao, F., Ho, Q., Dai, W., Wei, J., Zheng, X., Xing, E.P., Liu, T.Y. and Ma, W.Y., 2015, May. Lightlda: Big topic models on modest computer clusters. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 1351-1361). International World Wide Web Conferences Steering Committee.
- 92) Nguyen, D.Q., Billingsley, R., Du, L. and Johnson, M., 2015. Improving topic models with latent feature word representations. *Transactions of the Association for Computational Linguistics*, 3, pp.299-313.
- 93) Xie, P., Yang, D. and Xing, E., 2015. Incorporating word correlation knowledge into topic modeling. In *Proceedings of the 2015 conference of the north American chapter of the association for computational linguistics: human language technologies* (pp. 725-734).
- 94) Uehara, H., Ito, A., Saito, Y. and Yoshida, K., 2019, August. Prior-Knowledge-Embedded LDA with Word2vec—for Detecting Specific Topics in Documents. In *Pacific Rim Knowledge Acquisition Workshop* (pp. 115-126). Springer, Cham.
- 95) Zhang, Y., Chen, M., Huang, D., Wu, D. and Li, Y., 2017. iDoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization. *Future Generation Computer Systems*, 66, pp.30-35.
- 96) Koochi, H. and Kiani, K., 2016. User based collaborative filtering using fuzzy C-means. *Measurement*, 91, pp.134-139.
- 97) Majid, A., Chen, L., Chen, G., Mirza, H.T., Hussain, I. and Woodward, J., 2013. A context-aware personalized travel recommendation system based on geotagged social media data mining. *International Journal of Geographical Information Science*, 27(4), pp.662-684.
- 98) Jin, Y., Hu, M., Singh, H., Rule, D., Berlyant, M. and Xie, Z., 2010, September. MySpace video recommendation with map-reduce on qizmt. In *2010 IEEE Fourth International Conference on Semantic Computing* (pp. 126-133). IEEE.
- 99) Jiang, J., Lu, J., Zhang, G. and Long, G., 2011, July. Scaling-up item-based collaborative filtering recommendation algorithm based on hadoop. In *2011 IEEE World Congress on Services* (pp. 490-497). IEEE.
- 100) Wang, H., Lu, Y. and Zhai, C., 2010, July. Latent aspect rating analysis on review text data: a rating regression approach. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 783-792). ACM.
- 101) Park, M.H., Hong, J.H. and Cho, S.B., 2007, July. Location-based recommendation system using bayesian user's preference model in mobile devices. In *International conference on ubiquitous intelligence and computing* (pp. 1130-1139).

Springer, Berlin, Heidelberg.

- 102) Pazzani, M.J. and Billsus, D., 2007. Content-based recommendation systems. In *The adaptive web* (pp. 325-341). Springer, Berlin, Heidelberg.
- 103) Greer, C.F. and Ferguson, D.A., 2011. Using Twitter for promotion and branding: A content analysis of local television Twitter sites. *Journal of Broadcasting & Electronic Media*, 55(2), pp.198-214.
- 104) Li, Y., Yang, M. and Zhang, Z.M., 2013, October. Scientific articles recommendation. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management* (pp. 1147-1156). ACM.
- 105) Lin, K., Yang, H.F., Liu, K.H., Hsiao, J.H. and Chen, C.S., 2015, June. Rapid clothing retrieval via deep learning of binary codes and hierarchical search. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval* (pp. 499-502). ACM.
- 106) Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012.
- 107) Chiliguano, Paulo, and Gyorgy Fazekas. "Hybrid music recommender using content-based and social information." *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016.
- 108) Wu, S., Ren, W., Yu, C., Chen, G., Zhang, D. and Zhu, J., 2016, May. Personal recommendation using deep recurrent neural networks in NetEase. In *2016 IEEE 32nd International Conference on Data Engineering (ICDE)* (pp. 1218-1229). IEEE.
- 109) Shafqat, W.; Lee, S.; Malik, S.; Kim, H.C. The language of deceivers: Linguistic features of crowdfunding scams. In *Proceedings of the 25th International Conference Companion on World Wide Web, Montréal, QC, Canada, 11–15 April 2016*; pp. 99–100.
- 110) Kim, M.J.; Bonn, M.; Lee, C.K. The effects of motivation, deterrents, trust, and risk on tourism crowdfunding behavior. *Asia Pac. J. Tour. Res.* 2019, 25, 244–260.
- 111) Zenone, M.; Snyder, J. Fraud in medical crowdfunding: A typology of publicized cases and policy recommendations. *Policy Internet* 2019, 11, 215–234.
- 112) Hu, W.; Yang, R. Predicting the success of Kickstarter projects in the US at launch time. In *Proceedings of the SAI Intelligent Systems Conference, London, UK, 5–6 September 2019*; pp. 497-506.
- 113) Desai, N.; Gupta, R.; Truong, K. *Plead or Pitch? The Role of Language in Kickstarter Project Success*; Stanford University: Stanford, CA, USA, 2015.
- 114) Sawhney, K.; Tran, C. Tuason, R. *Using Language to Predict Kickstarter Success*; Stanford University: Stanford, CA, USA, 2016.
- 115) Westerlund, M.; Singh, I.; Rajahonka, M.; Leminen, S.; Can short-text project summaries predict funding success on crowdfunding platforms? In *ISPIM Conference Proceedings* (pp. 1-15). The International Society for Professional Innovation Management (ISPIM), 2019.
- 116) Do Carmo, R.A.; Kang, S.M.; Silva, R. Visualization of topic-sentiment dynamics in crowdfunding projects. In *Proceedings of the International Symposium on Intelligent Data Analysis, London, UK, 26–28 October 2017*; pp. 40–51.
- 117) Siering, M.; Koch, J.A.; Deokar, A.V. Detecting fraudulent behavior on crowdfunding platforms: The role of linguistic and content-based cues in static and dynamic contexts. *J. Manag. Inf. Syst.* 2016, 33, 421–455.
- 118) Cumming, D.J.; Hornuf, L.; Karami, M.; Schweizer, D. Disentangling crowdfunding from fraudfunding. SSRN 2016, doi:10.2139/ssrn.2828919.
- 119) Tran, T.; Lee, K.; Vo, N.; Choi, H.; Identifying on-time reward delivery projects

- with estimating delivery duration on kickstarter. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017, July, 2017, (pp. 250-257). ACM.
- 120) Lika, B., Kolomvatsos, K. and Hadjiefthymiades, S., 2014. Facing the cold start problem in recommender systems. *Expert Systems with Applications*, 41(4), pp.2065-2073.
- 121) Fernandez-Gago, C., Agudo, I. and Lopez, J., 2014. Building trust from context similarity measures. *Computer Standards & Interfaces*, 36(4), pp.792-800.
- 122) Jamali, M. and Ester, M., 2009, June. Trustwalker: a random walk model for combining trust-based and item-based recommendation. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 397-406). ACM.
- 123) Moradi, P. and Ahmadian, S., 2015. A reliability-based recommendation method to improve trust-aware recommender systems. *Expert Systems with Applications*, 42(21), pp.7386-7398.