SAVITRIBAI PHULE PUNE UNIVERSITY

A PROJECT REPORT ON

DEEP LEARNING FOR CONTENT BASED RETRIEVAL

BACHELOR OF ENGINEERING (Computer Engineering)

BY

Mandeep Singh	Exam No: B120224226
Pawan Kumar	Exam No: B120224233
Sachin Choudhary	Exam No: B120224240

Under the Guidance of Prof. (Dr.) S.R. DHORE

Onward to Glorv

DEPARTMENT OF COMPUTER ENGINEERING

Army Institute of Technology Alandi Road, Dighi Hills

Onward to Glorv

Army Institute of Technology, Pune

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the Project Entitled

DEEP LEARNING FOR CONTENT BASED RETRIEVAL

Submitted by

Mandeep Singh	Exam No: B120224226
Pawan Kumar	Exam No: B120224233
Sachin Choudhary	Exam No: B120224240

is a bonafide work carried out by Students under the supervision of Prof (Dr.) S.R. Dhore and it is submitted towards the fulfilment of the requirement of Bachelor of Engineering (Computer Engineering) Project.

Prof.(Dr.) S. R Dhore	Prof. (Dr.) S.R Dhore
Internal Guide	H.O.D
Dept. of Computer Engg.	Dept. of Computer Engg.

PROJECT APPROVAL SHEET

DEEP LEARNING FOR CONTENT BASED RETRIEVAL

is successfully completed by

Mandeep Singh	Exam No: B120224226
Pawan Kumar	Exam No: B120224233
Sachin Choudhary	Exam No: B120224240

At

DEPARTMENT OF COMPUTER ENGINEERING

(Army Institute of Technology)

SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

ACADEMIC YEAR 2017-2018

Prof.(Dr.) S. R Dhore	Prof.(Dr.) S.R Dhore
Internal Guide	H.O.D
Dept. of Computer Engg.	Dept. of Computer Engg.

ABSTRACT

Due to the perception of cheap publishing, organizations have been producing enormous content online since the hidden cost of maintenance and usability has always been neglected. This presents the opportunity for automatically maintaining crisp and usable content, especially in news articles in this project, we compared deep learning model and machine learning model using Naive-Bayes classifier and support Vector Machine(SVM) classifier to extract features from different classes of content and cluster them under an umbrella topic and remove information redundancy large in online news-store. For each news category, we then go on to predict popularity of documents using additional features based on the content only.

We conduct our experiments on different news corpuses and used online news from https://www.newsapi.org to fetch news and then classify them in four categories (Entertainment, Business, Technology, Medical). Our study also serves to remove information redundancy in the online news data-store.

ACKNOWLEDGEMENTS

*It gives us great pleasure in presenting the project report on **‘DEEP LEARNING FOR CONTENT BASED RETRIEVAL’**.*

*We would like to take this opportunity to thank my internal guide **Prof (Dr.) S.R. Dhore** for giving me all the help and guidance we needed. we are very grateful to them for their kind support. Their valuable suggestions were very helpful.*

In the end our special thanks to staff Member for providing various resources such as laboratory with all needed software platforms, continuous Internet connection, for our project.

Mandeep Singh
Pawan Kumar
Sachin Choudhary

INDEX

List of Figures

List of Tables

CHAPTER 1

SYNOPSIS

1.1 PROJECT TITLE

Deep Learning for Content Based Retrieval

1.2 PROJECT OPTION

Internal Project

1.3 INTERNAL GUIDE

Prof (Dr.) S.R. Dhore

1.4 TECHNICAL KEYWORDS

1 Deep learning for text
2 News
3 Redundancy
4 Content popularity
5 Text representation
6 Aggregation
7 Classification
8 Text Similarity
9 Pruning

1.5 PROBLEM STATEMENT

News sources now concentrate a large portion of attention on online mediums where they can disseminate their news effectively and to a large population. Due to the time-sensitive post aspect and intense competition for attention in the socially connected digital platform, accurately estimating the extent to which a news article will spread on the web is extremely valuable to journalists, content providers, advertisers, and news recommendation systems. However, predicting the online popularity of online news articles is a challenging task.

1.6 ABSTRACT

Due to the perception of cheap publishing, organizations have been producing enormous content online since the hidden cost of maintenance and usability has always been neglected. This presents the opportunity for automatically maintaining crisp and usable content, especially in news articles in this project , we compared deep learning model and machine learning model using Naive-Bayes classifier and support Vector Machine(SVM) classifier to extract features from different classes of content and cluster them under an umbrella topic and remove information redundancy large in online news-store. For each news category, we then go on to predict popularity of documents using additional features based on the content only.

We conduct our experiments on different news corpuses and used online news from https://www.newsapi.org to fetch news and then classify them in four categories (Entertainment, Business, Technology, Medical). Our study also serves to remove information redundancy in the online news data-store.

1.7 GOALS AND OBJECTIVES

Our main goal is extract useful content from news corpuses based on content.

- Remove redundant information
- Extract most useful news related to query

1.8 RELEVANT MATHEMATICS ASSOCIATED WITH THE PROJECT

System Description:

- Input: Dataset of news articles
- Output: most relevant news article based on specific category
- Success Conditions: If the system is able to extract most useful news article based on content and remove the redundant information
- Failure Conditions: If there is any redundant information in final output

1.9 PLAN OF PROJECT EXECUTION

Deliverables	Deadline	Candidate Responsible
Pre-processing	2nd July 2017	All Three
Deep Learning model	2nd August 2017	All Three
SVM & Naïve Bayes	17th August 2017	All Three
Obtain Comparison Results for classification models	15th September 2017	All Three
Detailed Literature Survey	10nd September 2017	All Three
Final Framework	10th October 2017	All Three
Paper submission	20th October 2017	All three
Preliminary Presentation	6th November 2017 20th December 2017	All three
Final Report	25th March 2018	All three

Table 1.2 Plan of project execution

CHAPTER 2

TECHNICAL KEYWORDS

2.1 AREA OF PROJECT

The area of project is Machine learning and Deep Learning, we use Naïve Bayes Classifier to extract features from large document corpus.

2.2 TECHNICAL KEYWORDS

1. Deep learning for text
2. Machine Learning
3. Naïve Bayes
4. Support Vector Machine
5. News
6. Redundancy
7. Content popularity
8. Text representation
9. Aggregation
10. Classification
11. Text Similarity
12. Pruning

CHAPTER 3

INTRODUCTION

3.1 PROJECT IDEA

Get the most relevant news from large corpus based on specific category without any redundant data

3.2 MOTIVATION OF THE PROJECT

News articles are very dynamic in nature due to continuously developing nature of the event and parallel reporting of the same, thus they have a very short span of life. The ease and low cost of online content creation and sharing have changed the traditional rules of competition for public attention. News sources now concentrate a large portion of attention on online mediums where they can disseminate their news effectively and to a large population.

Due to the time-sensitive post aspect and intense competition for attention in the socially connected digital platform, accurately estimating the extent to which a news article will spread on the web is extremely valuable to journalists, content providers, advertisers, and news recommendation systems.
However, predicting the online popularity of online news articles is a challenging task. First, context outside the web is often not readily accessible and elements such as local and geographical conditions and various circumstances that affect the population to make this prediction extremely difficult. Furthermore, network properties such as the structure of social networks that are propagating the news, influence variations among members and interplay between different sections of the web add other layers of complexity to this problem. Most significantly, intuition suggests that content of an article must play a significant role in its popularity. Content that resonates with most readers such as a major worldwide event can be expected to garner wide attention while specific content relevant only to a few may not be as successful. Content that is up-to-date and highlights all aspect of that article.

3.3 LITERATURE SURVEY

We read several IEEE papers which supports our project idea, review of these papers:

- **"Feature Selection based Classification using Naïve Bayes, J48 and Support Vector Machine",** Dipali Bhosale and Roshani Ade:

  The accuracy to improve accuracy of a classifier is to use the minimum number of features. This paper presents feature selection methods of co-relation based feature Selection, Wrapper method and Information Gain are used, before applying supervised learning based classification techniques.

- **"Finding news in Haystack - Event based news clustering with social media based ranking",** Martin Weber and Maarten H. Lamers**:**

  The coverage of news is no longer limited to certain number of pages or time, but with more news it is very difficult to find more relevant news on particular topic and due to this we get multiple articles of same topic in top stories.

  For example: Google News aggregates news from more than 25,000 sources (Cohen (2009)) and groups content automatically in sections. The front page shows the top stories of the day according to the user's geographical region, language, and settings. The full coverage of a story, resulting in dozens to thousands of similar stories can be shown. So, the better approach can be building a software that consists modular components for crawling, collecting, filtering, clustering, ranking and selecting news items that match the desired topic.

- **"The Pulse of News in Social Media: Forecasting Popularity",** Roja Bandari, SitaRam Asur, Bernando A. Huberman

  News articles are very dynamic due to their relation to continuously developing events that typically have short lifespans. News sources now concentrate a large portion of their attention on online mediums where they can disseminate their news effectively and to a large population. Most significantly, intuition suggests that the content of an article must play a crucial role in its popularity. Content that resonates with a majority of the readers such as a major world-wide event can be expected to garner wide attention while specific content relevant only to a few may not be as successful.

  This paper considers below 4 characteristics to judge news article importance:

  - The news source that generates and posts the article

- The category of news this article falls under
- The subjectivity of the language in the article
- Named entities mentioned in the article
- And then quantify these characteristics by a score making use of different scoring functions. We then use these scores to generate predictions of the spread of the news articles using regression and classification methods.

  The paper is successful in overall accuracy of 84% using classifiers.

- **"Character-level Convolutional Networks for Text Classification",** Xiang Xang, Junbo Zhao, Yann LeCun

  Text classification is a classic topic for natural language processing, in which one needs to assign predefined categories to free-text documents. The range of text classification research goes from designing the best features to choosing the best possible machine learning classifiers. To date, almost all techniques of text classification are based on words, in which simple statistics of some ordered word combinations (such as n-grams) usually perform the best.

  This paper only used a classification task as a way to exemplify ConvNets' ability to understand texts.

Paper Title	**Conclusion**
Feature Selection based Classification using Naïve Bayes, J48 and Support Vector Machine	Support Vector Machine with Information Gain and Wrapper method have better results.
Finding news in Haystack - Event based news clustering with social media based ranking	Selection and event-based clustering works well for content featured on the front page, it doesn't for individual topics. This makes it hard to find diverse coverage on custom topics.
The Pulse of News in Social Media: Forecasting Popularity	Use scores to generate predictions of the spread of the news articles using regression and classification methods.
Character-level Convolutional Networks for Text Classification	Treat text as a kind of raw signal at character level, and applying temporal (one-dimensional) ConvNets to it

Table 3.1 Comparison of Literature Survey

Other references:

- Using TF-IDF to Determine Word Relevance in Document Queries
- http://scikit-learn.org/dev/_downloads/scikit-learn-docs.pdf
- https://radimrehurek.com/gensim/models/doc2vec.html
- https://textblob.readthedocs.io/en/dev/
- https://stanfordnlp.github.io/CoreNLP/
- http://edutechwiki.unige.ch/en/Lingpipe
- https://tartarus.org/martin/PorterStemmer/
- https://docs.oracle.com/cd/B28359_01/text.111/b28304/astopsup.htm#CCREF1400
- http://www.vision.caltech.edu/anelia/DataPruning/
- http://www.kdnuggets.com/2015/03/deep-learning-text-understanding-from- scratch.html

CHAPTER 4

PROBLEM DEFINITION AND SCOPE

4.1 PROBLEM STATEMENT

News sources now concentrate a large portion of attention on online mediums where they can disseminate their news effectively and to a large population. Due to the time-sensitive post aspect and intense competition for attention in the socially connected digital platform, accurately estimating the extent to which a news article will spread on the web is extremely valuable to journalists, content providers, advertisers, and news recommendation systems. However, predicting the online popularity of online news articles is a challenging task.

4.1.1 Goals and objectives

Our main goal is extract useful content from news corpuses based on content.

- Remove redundant information
- Extract most useful news related to category

4.1.2 Statement of scope

A) Dataset

We make use of following datasets for our experiments and project execution:

1) **News Headlines of India** This dataset consists of 16 years of categorized headlines focused on India.
2) **News Aggregator Dataset** This dataset consists of headlines and categories of 400k news stories from 2014.

B) Baseline:

1) **News Aggregators** We conduct an internal survey to verify initial results of the pipeline when compared with different news agents and aggregators like Google News.

4.2 MAJOR CONSTRAINTS

- The software only considers content as major contributor as a extracting feature, and don’t consider impact of social media.

- Large number of news headlines may take large time for processing hence can reduce the performance and quality of feature extraction.

4.3 SCENARIO IN WHICH MULTICORE-CORE, EMBEDDED AND DISTRIBUTED COMPUTING USED

As the number of news is very large, multicore programming can be used to enhance the results and reduce the time taken for feature extraction.

4.4 OUTCOME

Top news based on content relevancy in different categories which are as follows:

- Top Stories
- Entertainment
- Health
- Business
- Technology

4.5 APPLICATIONS

- **Content caching and traffic management**: There is a hidden cost to publishing content, the cost to review and maintain the content. The millions of articles also affect the usability and maintainability of the site. In the long run, it is necessary to tackle redundant, outdated and trivial content which has been cursing the site.
- **Advertising:** This work can find its application in content-based advertisement alongside news pieces. It will optimize ad-placement logistics and revenues.
- **News Aggregation:** With our current event driven clusters knowledge base, we predict the popularity of written articles to be published in that domain. It will allow content writer to write more relevant and less redundant pieces that can make it different from current flowing articles. We have been aggregating up-to-date content rich articles ignoring social backlinks.
- **Trends and Forecasting:** Since the cache contains most popular pieces from different news events, we are able to show current trends with no redundancy. This data helps in forecasting future trends

4.6 HARDWARE RESOURCES REQUIRED

Sr. No	Parameter	Minimum Requirement	Justification
1	CPU	2 GHz	To operate on large datasets
2	RAM	3 GB	To operate on large Dataset

Table 4.1 Hardware Requirements

4.7 SOFTWARE RESOURCES REQUIRED

- Operating System: Ubuntu Gnome 3.20.4 64 bit
- Languages: Python
- Frameworks: Django
- Editor: Atom/Sublime

CHAPTER 5

PROJECT PLAN

5.1 PROJECT ESTIMATES

5.1.1 Reconciled Estimates

5.1.1.1 Cost Estimate

All the software used are open source software's. Extra cost might incur if the project is expanded further for more data-sets and heavy computations in memory and buying AWS services for the heavy computations.

5.1.1.2 Time Estimates

Estimated time of research is 4 months and project is 4 months.

5.1.2 Project Resources

1. Hardware: The minimum hardware specifications are as follows: CPU- Pentium dual core processor or higher
   RAM- 2GB ram or more Storage- 80GB hard disk
2. Software: Operating system: (64 bit- Windows 7 or higher, Linux 3.6 or higher) Python >= 2.7

5.2 RISK MANAGEMENT W.R.T. NP HARD ANALYSIS

This section discusses Project risks and the approach to managing them.

5.2.1 Risk Identification

For risks identification, review of scope document, requirements specifications and schedule is done. Answers to questionnaire revealed some risks. Each risk is categorized as per the categories mentioned in. Please refer table 5.1 for all the risks. You can have refereed following risk identification questionnaire.

1. Have top software and customer managers formally committed to support the project?
2. Are end-users enthusiastically committed to the project and the system/product to be built?

3. Are requirements fully understood by the software engineering team and its customers?
4. Have customers been involved fully in the definition of requirements?
5. Do end-users have realistic expectations?
6. Does the software engineering team have the right mix of skills?
7. Are project requirements stable?
8. Is the number of people on the project team adequate to do the job?

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Estimated Project Schedule	High	High	High	High
2	Project Scope Creep	Low	Low	Medium	Medium
3	Project Team Availability	Medium	Medium	High	High
4	Person Hours	High	High	High	High
5	Timeline Estimates Unrealistic	Medium	Medium	Medium	Medium
6	Poor Functional Match of Package to Initial System Requirements	Low	Low	low	Low

Table 5.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	> 75%
Medium	Probability of occurrence	26.75%
Low	Probability of occurrence is	< 25%

Table 5.2: Risk Probability definitions

Impact	Value	Description
Very high	> 10%	Schedule impact or Unacceptable quality
High	5 - 10%	Schedule impact or Some parts of the project have low Quality
Medium	< 5%	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 5.3: Risk I mpact definitions

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Estimated Project Schedule
Category	Development Environment.
Source	Software requirement Specification document.
Probability	High
Impact	High
Response	Mitigate
Strategy	Created comprehensive project timeline with frequent baseline reviews
Risk Status	Anticipated

Table 5.4: Risk Mitigation, Monitoring, Management

Risk ID	2
Risk Description	Project Scope Creep
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	Medium
Response	Mitigate
Strategy	Scope initially defined in project plan, reviewed monthly by project guide.

Risk Status	Identified

Risk ID	3
Risk Description	Project Team Availability
Category	Technology
Source	This was identified during early development and testing.
Probability	Medium
Impact	Very High
Response	Accept
Strategy	Continuous review of project momentum by all levels. If necessary, increase commitment by participants to full time status

Risk Status	Identified

5.3 PROJECT SCHEDULE

5.3.1 Project task set

Major Tasks in the Project stages are:

- Project Title Selection Discussion
- Proposal Submission
- Literature Survey
- Proposal Presentation
- Proposal Design
- Software Development Test Plan
- Testing & QA

5.3.2 Task network

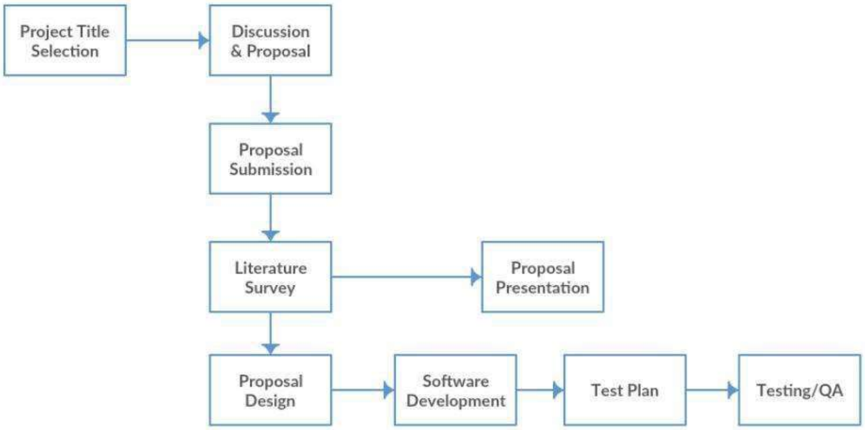


Fig 5.1 Task Network

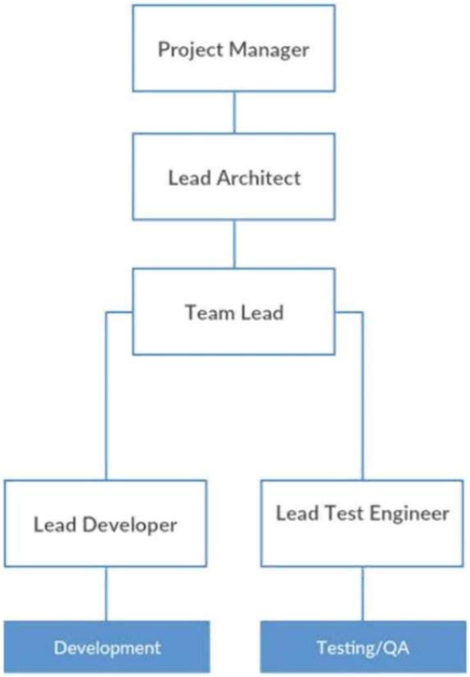


Figure 5.2: Team Structure

5.4 Management reporting and communication

Each team member is assigned a role and tasks to accomplish in the project. Regular team meetings at least twice a week.

Project records developed as any task is accomplished.

Team meetings with the project guide once a week reporting the progress of the project

CHAPTER 6

SOFTWARE REQUIREMENT SPECIFICATION

6.1 INTRODUCTION

6.1.1 Purpose and Scope of Document

Purpose of Our Project is to Cluster the Documents by using graph representation model. Scope of Our Project is to improve automatic document organization, topic extraction and fast information retrieval or filtering.

6.1.2 Overview of responsibilities of Developer

Dataset collection

We make use of following datasets for our experiments and project execution:

- **News Headlines of India** This dataset consists of 16 years of categorized headlines focused on India.
- **News Aggregator Dataset** This dataset consists of headlines and categories of 400k news stories from 2014.

Pre-processing

Pre-processing consisting of procedure that transforms the text data in the document to a structure templates for text mining. The main goal of processing is to get basic features or key terms from online

news text documents and enhance the relevancy between words and document and the relevancy between words and category. A document almost contains number of not necessary words that could negatively affect the clusters of the document. In this research using programming language python to execute the work. There are four steps that must be followed to clean up the text of phrases or words, namely: Split document into sentences, Stemming, Tokenization and Part of Speech Tagging.

- Stemming: Stemming is the process reduction of inflected words to their stem, root or base form or in general, a written word form. Stemming is used in determining domain vocabulary in domain analysis.

- Tokenization: It is the process of breaking a stream of text into words, phrases, symbols, or other meaningful elements called tokens. The list of tokens becomes the input data for further processing such as parsing or text mining. Tokenization is useful in linguistics, where it is a form of text segmentation, and in computer science, where it forms part of lexical analysis.

6.2 USAGE SCENARIO

This section provides various usage scenarios for the system to be developed.

6.2.1 User profiles

- Actors here are normal users.
- All the processing and result generation by application will be done by users.

6.2.2 Use-cases

Sr	Use Case	Description	Ac	Assumptions
1	Use Case 1	based on the classification of data of given datasets the	us ers	users have all set of privilege

Table 6.1: Use Cases

6.2.3 Use Case View

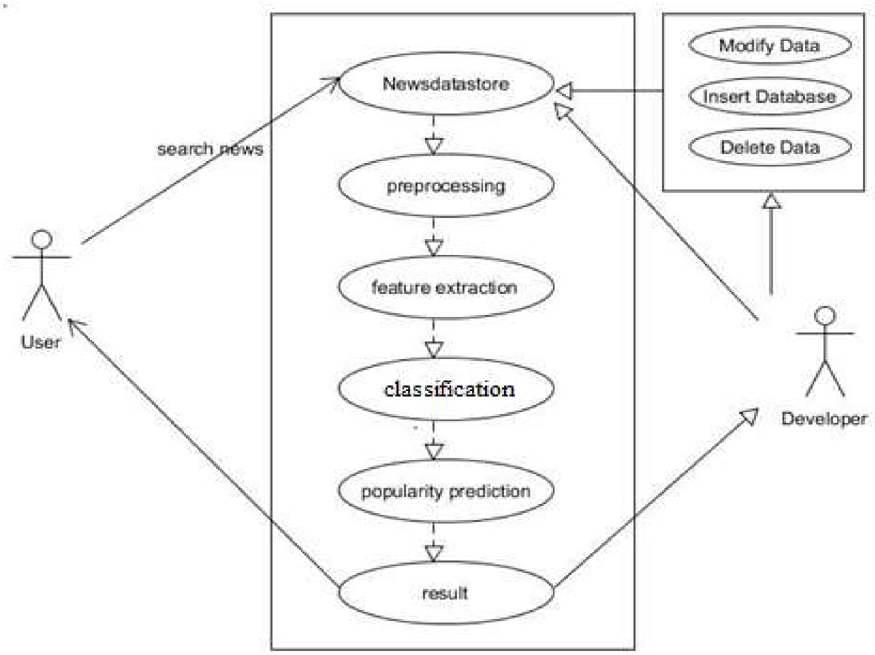


Figure 6.1: Use case diagram

6.3 DATA MODEL AND DESCRIPTION

6.3.1 Data Description

- **News Headlines of India** This dataset consists of 16 years of categorized headlines focused on india.
- **News Aggregator Dataset** This dataset consists of headlines and categories of 400k news stories from 2014.

6.3.2 Data objects and Relationships

All the data-sets have different number of documents which will be pre-processed and then used for feature extraction.

6.4 FUNCTIONAL MODEL AND DESCRIPTION

6.4.1 Data Flow Diagram

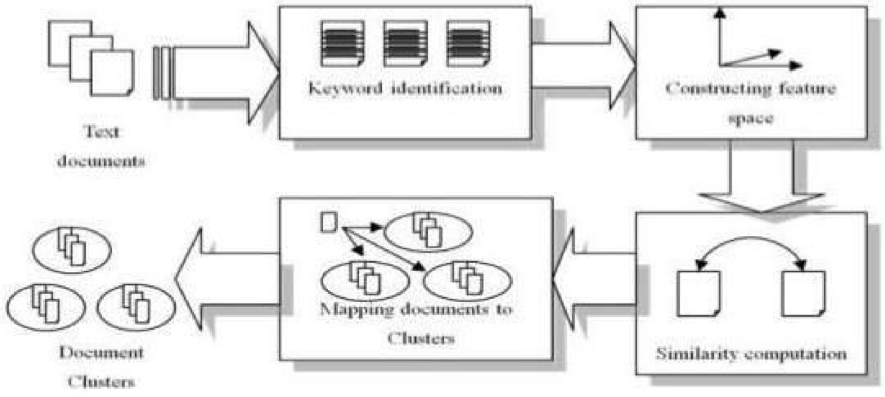


Fig. 6.2 Data Flow Diagram

A data flow diagram is a graphical representation of “flow” of data through an information system, modelling its process aspects. DFD gives overview of system without going into much details.

It specifies input and output to the system, and how data advance through the system.

In our project, input will be set of news headlines fetched from https://www.newsapi.org and output will be most relevant news based on content according to specific category

6.4.2 Activity Diagram:

- The Activity diagram represents the steps taken.

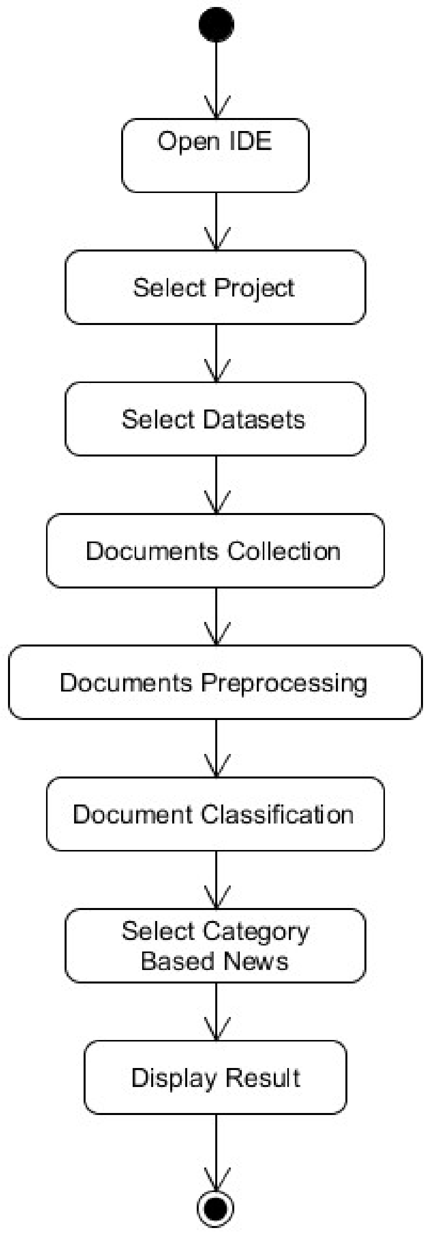


Fig 6.3: Activity Diagram

6.4.3 Non Functional Requirements:

- Interface Requirements

1. Query reports
2. Keyboard

- Performance Requirements

1. Response time of data-set loading should be optimized.
2. Time for query evaluation should be optimized.

- Software quality attributes

1. Reliability
2. Modifiability
3. Adaptability
4. Correctness
5. Robustness
6. Testability
7. Flexibility

State Diagram:

Fig.6.3 example shows the state transition diagram of Document Clustering. The states are represented in ovals and state of system gets changed when certain events occur. The transitions from one state to the other are represented by arrows. The Figure shows important states and events that occur while creating new project.

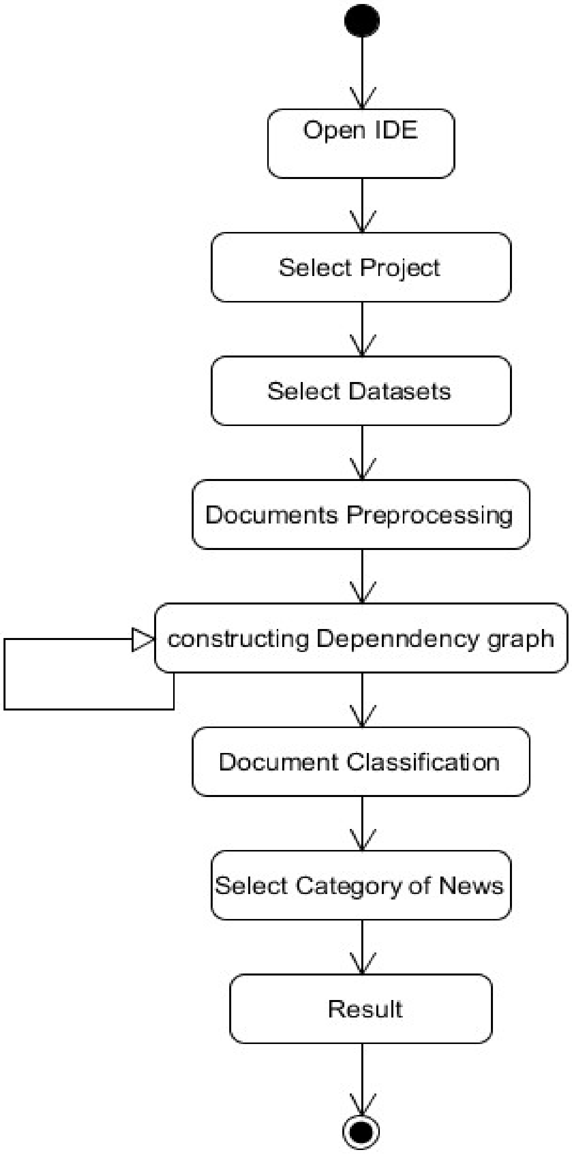


Figure 6.4: State transition diagram

6.4.5 Design Constraints

- Data extraction timing.
- Pre-processing error rate
- Removing redundant information

6.4.6 Software Interface Description

- User Interface for loading the data-sets in project.
- User enters query
- System computes top news according to category (entertainment, business, technology, medical)

CHAPTER 7

DETAILED DESIGN DOCUMENT USING APPENDIX A AND B

7.1 SOLUTION FRAMEWORK

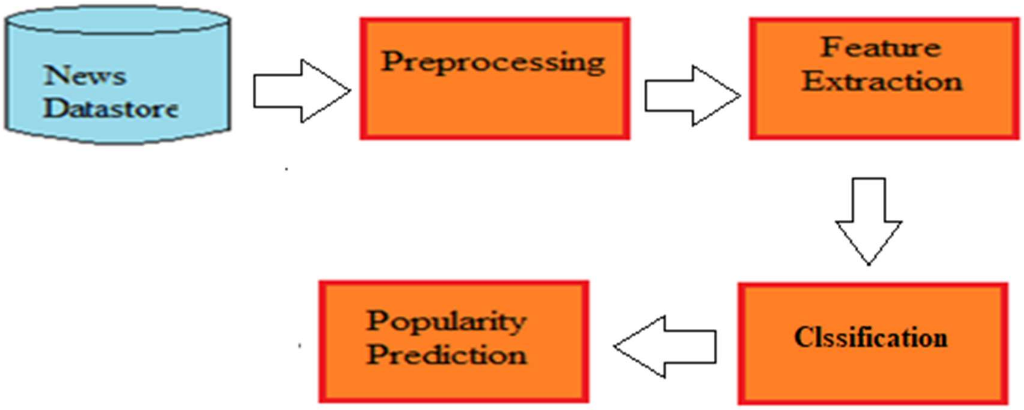


Fig. 7.1 Solution framework of project

7.1.1 PREPROCESSING

We pre-process the data to make processing more meaningful Pre-processing steps:

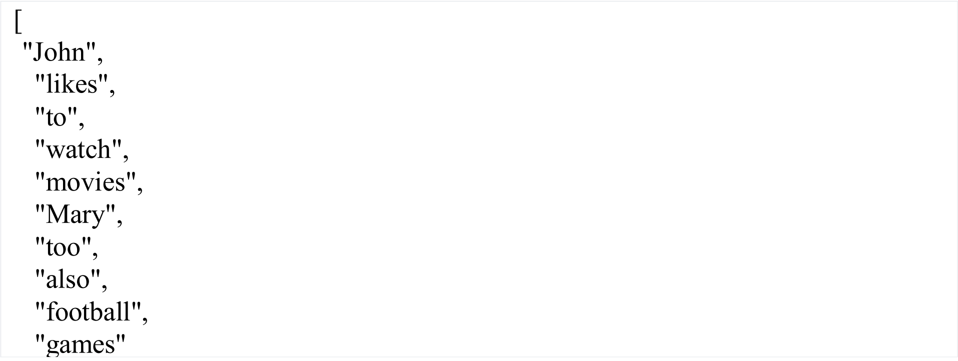
- **Filtering:** Removal of mark-up, punctuation and special characters from sentences.
- **Tokenization:** Splitting of text into individual chunk.
- **Stemming:** Reduction of words to their base form.
- **Stop words removal:** Deletion of words that do not convey any special meaning, use stop word list by oracle.
- **Pruning:** Removal of words that do appear with a low frequency throughout the text.

The result of these pre-processing steps is a set of feature words.

Example of Tokenization:

Document 1: John likes to watch movies. Mary likes movies too. Document 2: John also likes to watch football games.

Based on the above two document, the list can be constructed as using bag of words approach:



```
[
 "John",
   "likes",
   "to",
   "watch",
   "movies",
   "Mary",
   "too",
   "also",
   "football",
   "games"
```

Example of stemming:

The words “stem”, “stemmer”, “stemming”, “stemmed” are based on root word “stem”.

7.1.2 TEXT UNDERSTANDING

Text understanding consists in reading texts formed in natural languages, determining the explicit or implicit meaning of each element such as words, phrases, sentences, and paragraphs, and making inferences about the implicit or explicit properties of these texts. Text understanding can be handled by a deep learning system without artificially embedding knowledge about words, phrases, sentences or any other syntactic or semantic structures associated with a language.

ConvNets for text understanding are modular, where gradients are obtained by back-propagation to perform optimization.

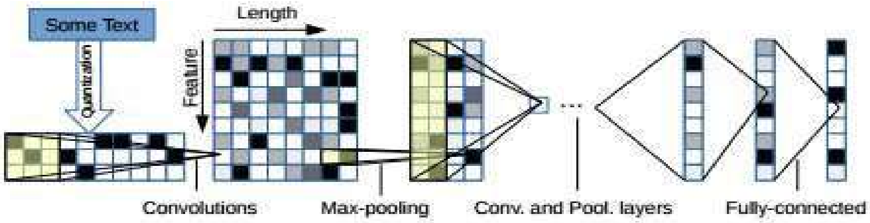


Fig. 7.2 Deep CovNets model illustration for Feature ExtractioN

Key Modules: It is a temporal convolutional module, which simply computes a 1- D convolution between input and output.

7.1.3 CLASSIFICATION MODEL

7.1.3.1 Naïve Bayes Classifier

Naive bayes classifier is probabilistic classifier based on Naives Bayes theorem. This model assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set.

$$\underset{\text{Posterior Probability}}{P(c\mid x)} = \frac{\overset{\text{Likelihood}}{P(x\mid c)}\,\overset{\text{Class Prior Probability}}{P(c)}}{\underset{\text{Predictor Prior Probability}}{P(x)}}$$

$$P(c\mid \mathrm{X}) = P(x_1\mid c)\times P(x_2\mid c)\times\cdots\times P(x_n\mid c)\times P(c)$$

7.1.3.2 SVM Classifier

SVM is a short form for Support Vector Machine. In machine learning, SVM is a supervised learning model with associate learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier (although methods such as Platt scaling exist to use SVM in a probabilistic classification setting). An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

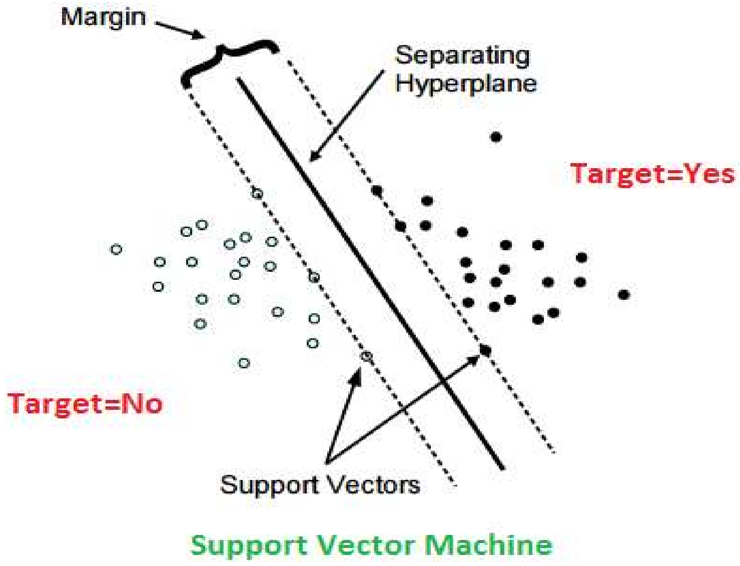


Fig. 7.3 Support Vector Machine (SVM)

7.1.4 POPULARITY PREDICTION:

We are motivated to predict popularity of article beforehand only from content based features and select only relevant articles of specific category. We intend to generate a score for each of the articles unlike categorizing them into classes.

Features:

The choice of features is motivated by multiple questions. Does the source agent reach many readers? Does the language connect with the reader? Has the article became outdated? Do we have some information in the news piece or not? Is the news worthy of a read? These questions helped us in designing following five features.

- **Age:** The date of publication of news given by the dataset. We remove few records with missing dates.
- **Text Quality:** The ratio of size of document before and after pre-processing.
- **Source Quality:** The popularity of source of the content given by initial number of hits provided by the source. If missing, we use the popularity of news agent as a whole. This is log-normalized to account for high range of hits or we make use of Alexa-ranking of website.
- **Subjectivity:** This examines whether an article is written in more emotional, touchy tone, where it connects with the reader.

7.1.5 SELECTION ALGORITHM: TF-IDF

TF-IDF stands for “Term Frequency – Inverse Document Frequency”. It is way to score the importance of words (or “terms”) in a document based on how frequently they appear across multiple documents.

It can be explained as follows:

- If a word appears frequently in a document, it’s important. Give the word a high score.
- But if the word appears in many documents, it’s not a unique identifier. Give the word a low score.

Therefore, common words like “the” and “and”, which appear in many documents, will be scaled down. Words that appear frequently in a single document will be scaled up.

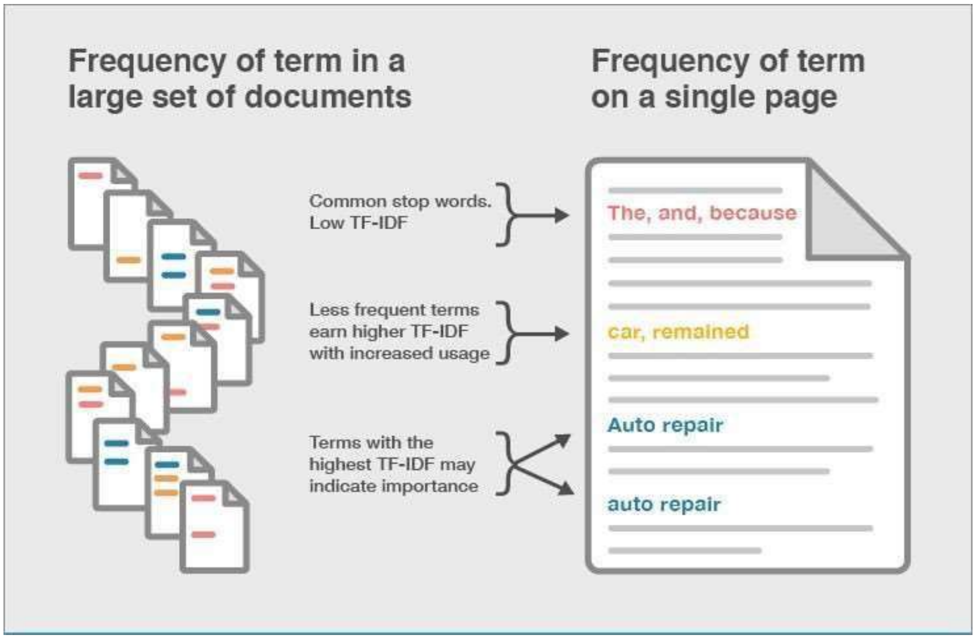

Frequency of term in a large set of documents
Frequency of term on a single page
Common stop words. Low TF-IDF
The, and, because
Less frequent terms earn higher TF-IDF with increased usage
car, remained
Terms with the highest TF-IDF may indicate importance
Auto repair
auto repair

Fig. 7.4 TF-IDF algorithm

For example, in above picture from the large document corpus, words like “the”, “and”, “because” have low tf-idf. Then there might be some words which have intermediate tf-idf value, the words which have highest tf-idf value like “auto repair”, “Auto repair” indicate importance and they can be processed as result.

7.2 COMPONENT DESIGN

7.2.1 Class Diagram

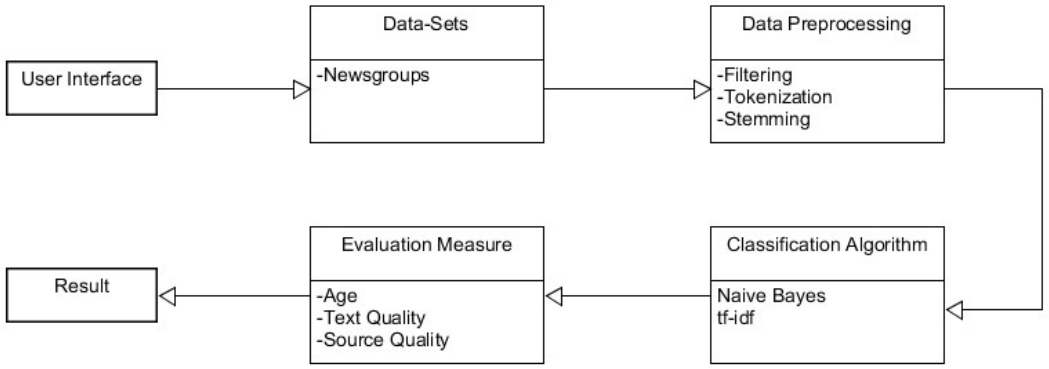


Fig 7.5 Class Diagram

Class diagram is a type of static structure diagram that describes the structure of the system by showing the system's classes, their attributes, operations and the relationship among the objects.

It is divided into user interface, where user can see the results, the dataset class contains all the offline news data, data pre-processing includes pre-processing steps such as filtering, tokenization and stemming, and for further classification Naïve Bayes Classifier is used. Final evaluation measure is based age, text quality and source quality and finally the most relevant documents are displayed.

CHAPTER - 8
IMPLEMENTATION

8.1 INTRODUCTION

The Project Implemented in python-Django framework in which user can select a category of news(Top Stories, Entertainment, Medical, Technology, Business) and headlines of that particular category are displayed on web browser based on content relevancy.

8.2 TOOLS AND TECHNOLOGY USED

1. **Python**

   Python is an Interpreted high-level programming language for genera-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales.

2. **Django (Web Framework)**

   Django is a free and open-source web framework, written in Python, which follows the model-view-template (MVT) architectural pattern. It is maintained by the Django Software Foundation (DSF), an independent organization established as a 501(c)(3) non-profit.

   Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and "pluggability" of components, less code, low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

3. **Atom (text editor)**

Atom is a free and open-source text and source code editor for macOS, Linux, and Microsoft Windows with support for plug-ins written in Node.js, and embedded Git Control, developed by GitHub. Atom is a desktop application built using web technologies. Most of the extending packages have free software licenses and are community-built and maintained. Atom is based on Electron (formerly known as Atom Shell), a framework that enables cross-platform desktop applications using Chromium and Node.js. It is written in Coffee Script and Less.

8.3 METHODOLOGIES/ALGORITHM DETAILS

Training the Model:

We tried three classifiers I.e. Naïve Bayes classifier, SVM classifier and deep learning model using Keras, out of all three we got best accuracy percentage for Naïve Bayes Classifier (89.64 %).

```
def train_model(news, keywords):
news['category'] = news['category'].fillna('x')
vectorizer = CountVectorizer()
x = vectorizer.fit_transform(news['title'])
x = TfidfTransformer().fit_transform(x)
encoder = LabelEncoder()
y = encoder.fit_transform(news['category'])
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2)

# classification using naive bayes
nb = MultinomialNB()
nb.fit(x_train, y_train)
# classification using svm classifier
text_clf_svm = SGDClassifier(loss='hinge', penalty='l2',alpha=1e-3, max_iter=5, random_state=42)
text_clf_svm = text_clf_svm.fit(x_train, y_train)

```

```
print("Training complete")
print("svm Accuracy:: ",text_clf_svm.score(x_test,y_test))
print("Naive bayes Accuracy ", nb.score(x_test, y_test))

# Now save the created model
pickle.dump(vectorizer, open("pickle-data/vectorizer.p", "wb"))
pickle.dump(encoder, open("pickle-data/encoder.p", "wb"))
pickle.dump(keywords, open("pickle-data/keywords.p", "wb"))
pickle.dump(nb, open("pickle-data/classifier.p", "wb"))
print("Model saved")
```

Top keywords obtained for each category:

category: Business (b)

top 10 keywords: [('us', 163), ('stocks', 91), ('china', 88), ('new', 84), ('says', 59), ('prices', 53), ('sales', 53), ('data', 52), ('market', 51), ('billion', 49)

category: Entertainment (e)

top 10 keywords: [('new', 874), ('kim', 611), ('kardashian', 597), ('video', 585), ('season', 536), ("'the", 481), ('review', 440), ('movie', 440), ('star', 418), ('thrones', 391)

category: Technology (t)

top 10 keywords: [('google', 1136), ('apple', 997), ('new', 814), ('samsung', 776), ('microsoft', 677), ('galaxy', 578), ('facebook', 560), ('one', 389), ('android', 386), ('us', 332)]

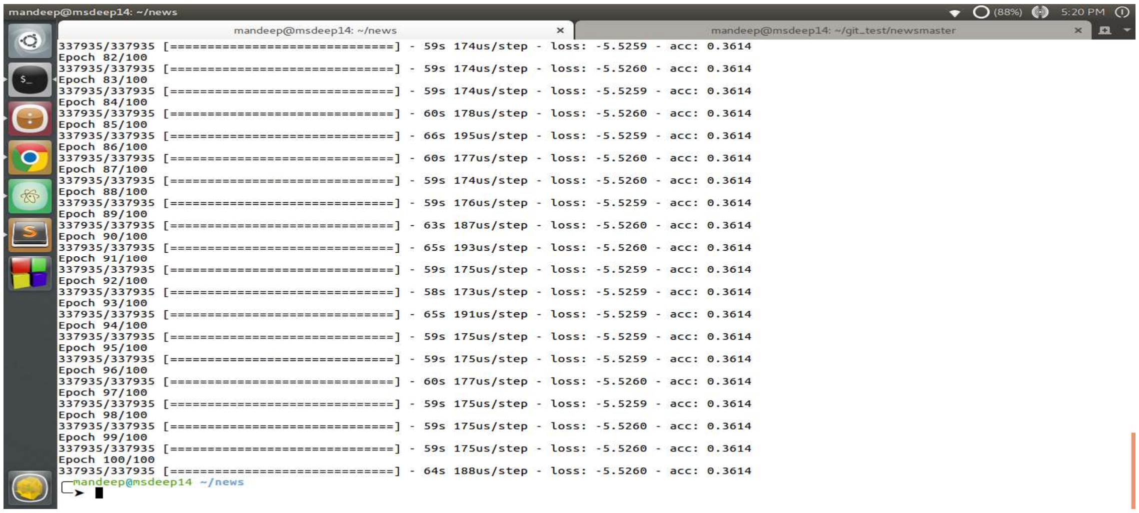
category: Medical (m)

top 10 keywords: [('ebola', 370), ('study', 345), ('health', 285), ('cancer', 263), ('new', 257), ('may', 209), ('mers', 198), ('virus', 194), ('us', 183), ('risk', 174)]

Accuracy obtained by different models:

Model used	Accuracy
SVM MODEL	0.865
DEEP LEARNING MODEL	0.896
NAÏVE BAYES CLASSIFICATION	0.361

Table. 8.1 Accuracy of models



```
337935/337935 [==============================] - 59s 174us/step - loss: -5.5259 - acc: 0.3614
Epoch 82/100
337935/337935 [==============================] - 59s 174us/step - loss: -5.5260 - acc: 0.3614
Epoch 83/100
337935/337935 [==============================] - 59s 174us/step - loss: -5.5259 - acc: 0.3614
Epoch 84/100
337935/337935 [==============================] - 60s 178us/step - loss: -5.5260 - acc: 0.3614
Epoch 85/100
337935/337935 [==============================] - 66s 195us/step - loss: -5.5259 - acc: 0.3614
Epoch 86/100
337935/337935 [==============================] - 60s 177us/step - loss: -5.5260 - acc: 0.3614
Epoch 87/100
337935/337935 [==============================] - 59s 174us/step - loss: -5.5260 - acc: 0.3614
Epoch 88/100
337935/337935 [==============================] - 59s 176us/step - loss: -5.5259 - acc: 0.3614
Epoch 89/100
337935/337935 [==============================] - 63s 187us/step - loss: -5.5260 - acc: 0.3614
Epoch 90/100
337935/337935 [==============================] - 65s 193us/step - loss: -5.5260 - acc: 0.3614
Epoch 91/100
337935/337935 [==============================] - 59s 175us/step - loss: -5.5259 - acc: 0.3614
Epoch 92/100
337935/337935 [==============================] - 58s 173us/step - loss: -5.5259 - acc: 0.3614
Epoch 93/100
337935/337935 [==============================] - 65s 191us/step - loss: -5.5259 - acc: 0.3614
Epoch 94/100
337935/337935 [==============================] - 59s 175us/step - loss: -5.5259 - acc: 0.3614
Epoch 95/100
337935/337935 [==============================] - 59s 175us/step - loss: -5.5259 - acc: 0.3614
Epoch 96/100
337935/337935 [==============================] - 60s 177us/step - loss: -5.5260 - acc: 0.3614
Epoch 97/100
337935/337935 [==============================] - 59s 175us/step - loss: -5.5259 - acc: 0.3614
Epoch 98/100
337935/337935 [==============================] - 59s 175us/step - loss: -5.5260 - acc: 0.3614
Epoch 99/100
337935/337935 [==============================] - 59s 175us/step - loss: -5.5260 - acc: 0.3614
Epoch 100/100
337935/337935 [==============================] - 64s 188us/step - loss: -5.5260 - acc: 0.3614
┌mandeep@msdeep14 ~/news
└➤ █
```

Fig. 8.1 Accuracy for deep learning model

Find the most relevant news articles:

Relevancy of article is decided on below factors:

- Age: The date of publication of news given by the dataset.
- Source Quality: The popularity of source of the content given by initial number of hits provided by the source. If missing, we use the popularity of news agent as a whole. This is log-normalized to account for high range of hits or we make use of Alexa-ranking of website.
- Subjectivity: This examines whether an article is written in more emotional, touchy tone, where it connects with the reader.
- Text Quality:
- Named Entities: We hypothesize that well-known named entities will cause a further spead of the article.

```
def get_relevant_articles(article_dict, category):
     # sort the articles according to source
    relevant_articles = {}
    for article, article_value in article_dict.items():
        relevant_articles[article] = get_score(article, article_value, category)

    sorted_articles = sorted(relevant_articles.items(), key=operator.itemgetter(1))
    # sort article_dict according to sorted_articles
    sorted_article_dict = OrderedDict()
    for ar in reversed(sorted_articles):
        sorted_article_dict[ar[0]] = article_dict[ar[0]]
  return sorted_article_dict
```

Following method gives the combined score of all the relevant factors stated above.

```
def get_score(article, article_value, category):
    dt = article_value[5].split('-')
    dt1 = date(int(dt[0]), int(dt[1]), int(dt[2]))
    dt2 = datetime.date.today()
    age_score = (dt2 - dt1).days
    source_score = get_source_core(article_value, category)
    sentiment_score = get_sentiment_score(article_value[3])
    text_quality_score = get_text_quality_score(article_value[3])
    named_entities = get_named_entities(article_value[3])
    return (age_score + source_score + sentiment_score + text_quality_score)
```

We defined source score based on number of hits a website get.

```
technology = {'techcrunch':10, 'the-next-web':9, 'wired':8, 'mashable':7, 'the-
verge':6,'techradar':5,'business-insider':4, 'ars-technica':4,'engadget':3,'recode':2}
business = {'business-insider':10, 'bloomberg':9, 'the-wall-treet-
journal':8, 'cnbc':7, 'financial-times':6,'financial-post':5,'australian-financial-
review':4,'fortune':3,'the-economist':2}
entertainment = {'entertainment-weekly':10, 'mtv-news':9, 'mtv-news-uk':8,'vice-
news':7,'buzzfeed':6,
'daily-mail':5,'the-lad-bible':4,'polygon':3}
medical = {'medical-news-today':10}
```

Following method gives sentiment score of the articles, whether it is positive, negative or neutral.

```
def get_sentiment_score(sentence):
    sid = SentimentIntensityAnalyzer()
    ss = sid.polarity_scores(sentence)
    try:
        return ss['compound']
    except:
        return 0
# Following method gives text quality of article.
def get_text_quality_score(sentence):
   if sentence == '':
       return 0
   else:
       prev_score = len(sentence.split(' '))
       keywords = []
       after = preprocessing.normalize_text(sentence, keywords)
       after_score = len(after.split(' '))
       # print("prev, after score:: ", prev_score, "  ", after_score)
   return (after_score/prev_score)
```

8.4 VERIFICATION AND VALIDATION FOR ACCEPTANCE

Verification

Verification is the process of evaluating products of a development phase to find out whether they meet the specified requirements.

The objective of Verification is to make sure that the product being develop is as per the requirements and design specifications. So reviews were done on the code and regular meeting were done

Validation

Validation is the process of evaluating software at the end of the development process to determine whether software meets the customer expectations and requirements. Testing was done on the system to make sure it is correct.

CHAPTER - 9

SOFTWARE TESTING

9.1 TYPE OF TESTING USED

- **Unit Testing**: Unit testing is the testing of an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit he/she has implemented is producing expected output against given input.
- **Integration Testing**: Integration testing is testing in which a group of com-ponents are combined to produce output. Also, the interaction between soft-ware and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing.
- **Functional Testing**: Functional testing is the testing to ensure that the speci-fied functionality required in the system requirements works. It falls under the class of black box testing.
- **System Testing**: System testing is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing

9.2 TEST CASES AND TEST RESULTS

1. Unit Testing

Short Description: Check if each unit is working properly

Test ID	Steps	Expected Result	Actual Result	Pass/ Fail
T001	1.Give training string as input to vectorizer 2.Then transform testing array	The array is taken and divided into individual words	The string is vectorized	Pass
T002	1.Give raw data as input to normalize_text function	The non-ASCII characters are removed	The data is clean for further processing	Pass
T003	1.Give processed matrix to classifier functions	The classifiers classify the data with the help of the model and give out accuracy and time taken	The classifiers give result in form of the Subject class.	Pass
T004	1.Store the results from classifiers in an array and print the result	Print the metrics in tabular form	Tabular form of results obtained	Pass

Table 9.1: Unit Testing

2. System Testing

Short Description: Check how system is working

Test ID	Steps	Expected Result	Actual Result	Pass/ Fail
T001	1.Give input of text Headline from	Accuracy of different classifiers and time taken should be	The results are obtained	Pass

	Document	reported		
T002	1.Try to implement the Different virtual environme system On nt	The system should give the same result irrespective of platform	Same result	Pass

Table 9.2: System Testing

3. Integration Testing

Short Description: Check how units are working together

Test ID	Steps	Expected Result	Actual Result	Pass/ Fail
T001	1.Give training string as input to vectorizer 2.Give result to tf-idf vectorizer	The string is taken and vectorized to tf-idf form	The string is vectorized	Pass
T002	1.Give raw data as input to normalizetext function 2.Give result to count vectorizer	The non ASCII characters are removed and data is vectorized	The data is vectorized	Pass
T003	1.Give processed matrix to classifier functions 2.Store result in array form	The classifiers classify the data and print the results	The results are printed	Pass

Table 9.3: Integration Testing

4. Functional Testing

Short Description: Check if units are functioning properly.

Test ID	Steps	Expected Result	Actual Result	Pass/ Fail
T001	1.Give training text as input to vectorizer	A vector array of counts of each unique word should be formed	The text is vectorized.	Pass
T002	1.Give raw data as input to normalizetext function	The non-ASCII characters are removed	The data is clean for further processing	Pass
T003	1.Give processed matrix to classifier functions	The classifiers classify the data and give results	Accuracy and time taken are obtained	Pass
T004	1.Store the results from classifiers in an array and print the result	Print the metrics in tabular form	Tabular form of results not obtained	Pass

Table 9.4: Functional Testing

CHAPTER -10

RESULT

RESULTS AND EVALUATION:

Bar graph showing number of news headlines in each category created using matplotlib

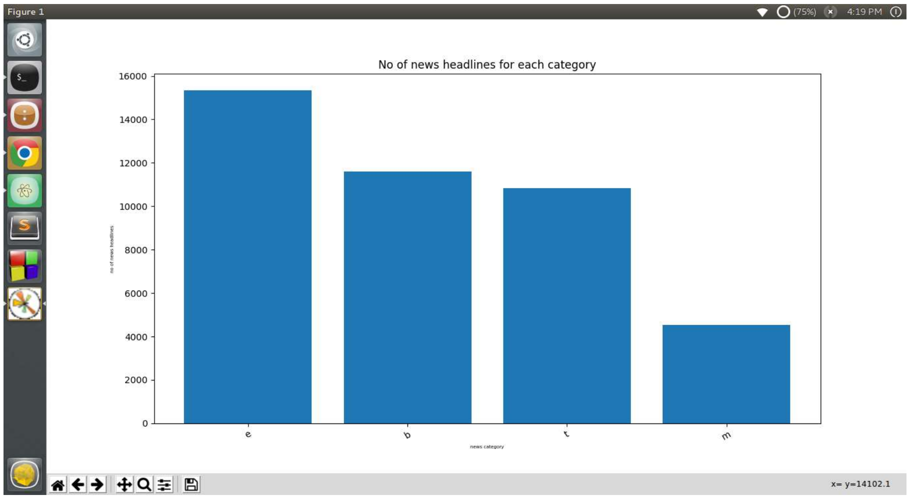
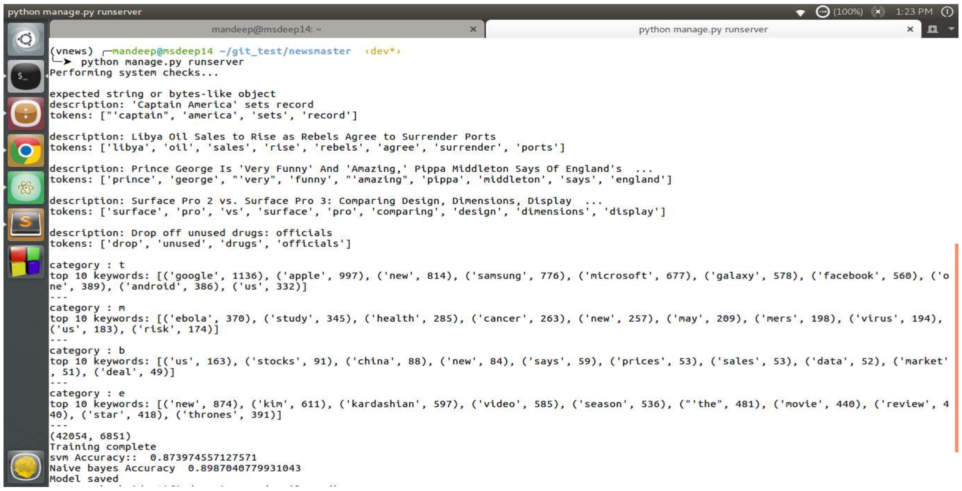


Fig.10.1 Number of news headlines in each category

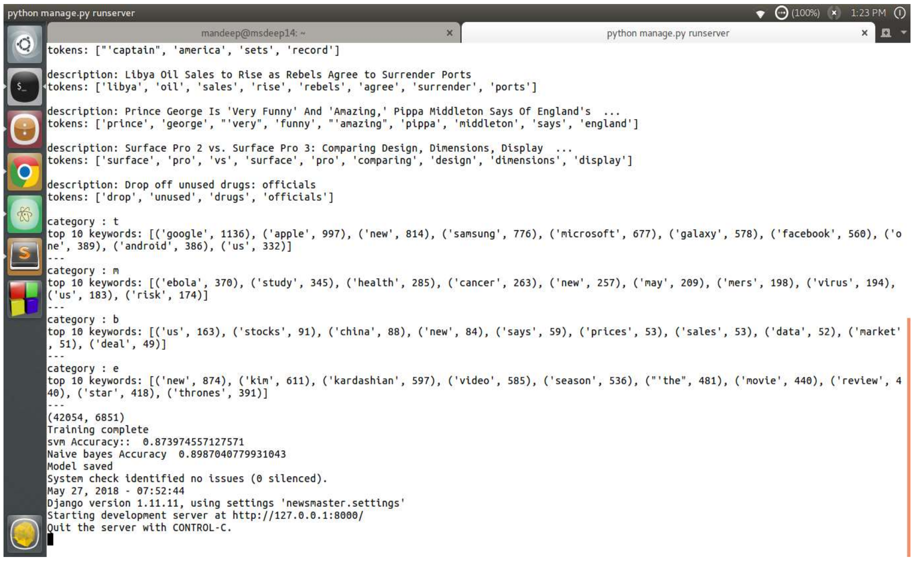
A model is already created in the project, for recreating the model, uncomment calling of method perform_preprocessing() in newsapp/preprocessing.py

The results obtained on backend server due to pre-processing steps are shown in below figures.



```
(vnews) ┌─mandeep@msdeep14 ~/git_test/newsmaster  ‹dev*›
└─➤ python manage.py runserver
Performing system checks...

expected string or bytes-like object
description: 'Captain America' sets record
tokens: ["'captain", 'america', 'sets', 'record']

description: Libya Oil Sales to Rise as Rebels Agree to Surrender Ports
tokens: ['libya', 'oil', 'sales', 'rise', 'rebels', 'agree', 'surrender', 'ports']

description: Prince George Is 'Very Funny' And 'Amazing,' Pippa Middleton Says Of England's  ...
tokens: ['prince', 'george', "'very", 'funny', "'amazing", 'pippa', 'middleton', 'says', 'england']

description: Surface Pro 2 vs. Surface Pro 3: Comparing Design, Dimensions, Display  ...
tokens: ['surface', 'pro', 'vs', 'surface', 'pro', 'comparing', 'design', 'dimensions', 'display']

description: Drop off unused drugs: officials
tokens: ['drop', 'unused', 'drugs', 'officials']

category : t
top 10 keywords: [('google', 1136), ('apple', 997), ('new', 814), ('samsung', 776), ('microsoft', 677), ('galaxy', 578), ('facebook', 560), ('o
ne', 389), ('android', 386), ('us', 332)]
---
category : m
top 10 keywords: [('ebola', 370), ('study', 345), ('health', 285), ('cancer', 263), ('new', 257), ('may', 209), ('mers', 198), ('virus', 194),
('us', 183), ('risk', 174)]
---
category : b
top 10 keywords: [('us', 163), ('stocks', 91), ('china', 88), ('new', 84), ('says', 59), ('prices', 53), ('sales', 53), ('data', 52), ('market'
, 51), ('deal', 49)]
---
category : e
top 10 keywords: [('new', 874), ('kim', 611), ('kardashian', 597), ('video', 585), ('season', 536), ("'the", 481), ('movie', 440), ('review', 4
40), ('star', 418), ('thrones', 391)]
---
(42054, 6851)
Training complete
svm Accuracy::  0.873974557127571
Naive bayes Accuracy  0.8987040779931043
Model saved
```

Fig 10.2 Headline tokenized into words and top keywords for each news category

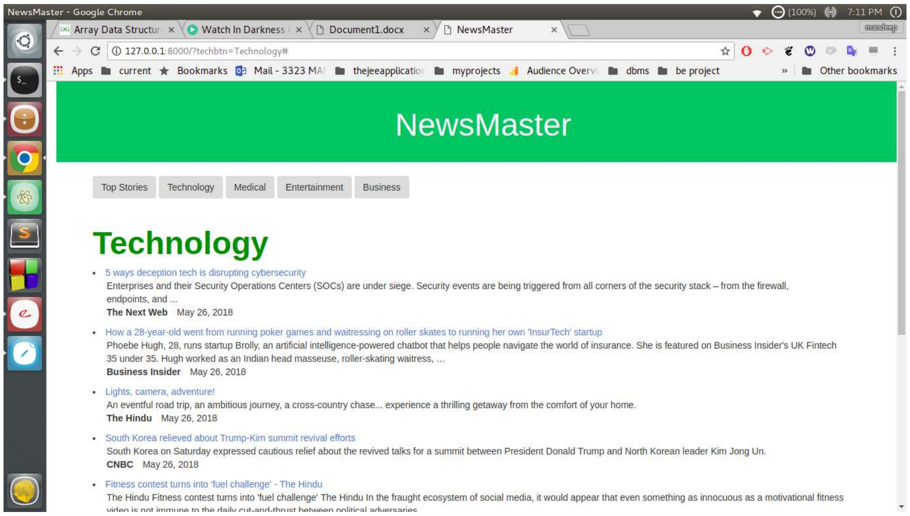


```
tokens: ["'captain", 'america', 'sets', 'record']

description: Libya Oil Sales to Rise as Rebels Agree to Surrender Ports
tokens: ['libya', 'oil', 'sales', 'rise', 'rebels', 'agree', 'surrender', 'ports']

description: Prince George Is 'Very Funny' And 'Amazing,' Pippa Middleton Says Of England's  ...
tokens: ['prince', 'george', "'very", 'funny', "'amazing", 'pippa', 'middleton', 'says', 'england']

description: Surface Pro 2 vs. Surface Pro 3: Comparing Design, Dimensions, Display  ...
tokens: ['surface', 'pro', 'vs', 'surface', 'pro', 'comparing', 'design', 'dimensions', 'display']

description: Drop off unused drugs: officials
tokens: ['drop', 'unused', 'drugs', 'officials']

category : t
top 10 keywords: [('google', 1136), ('apple', 997), ('new', 814), ('samsung', 776), ('microsoft', 677), ('galaxy', 578), ('facebook', 560), ('o
ne', 389), ('android', 386), ('us', 332)]
---
category : m
top 10 keywords: [('ebola', 370), ('study', 345), ('health', 285), ('cancer', 263), ('new', 257), ('may', 209), ('mers', 198), ('virus', 194),
('us', 183), ('risk', 174)]
---
category : b
top 10 keywords: [('us', 163), ('stocks', 91), ('china', 88), ('new', 84), ('says', 59), ('prices', 53), ('sales', 53), ('data', 52), ('market'
, 51), ('deal', 49)]
---
category : e
top 10 keywords: [('new', 874), ('kim', 611), ('kardashian', 597), ('video', 585), ('season', 536), ("'the", 481), ('movie', 440), ('review', 4
40), ('star', 418), ('thrones', 391)]
---
(42054, 6851)
Training complete
svm Accuracy::  0.873974557127571
Naive bayes Accuracy  0.8987040779931043
Model saved
System check identified no issues (0 silenced).
May 27, 2018 - 07:52:44
Django version 1.11.11, using settings 'newsmaster.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

Fig.10.3 Accuracy of SVM and Naïve Bayes classifier

Project NewsMaster screenshots on Browser:

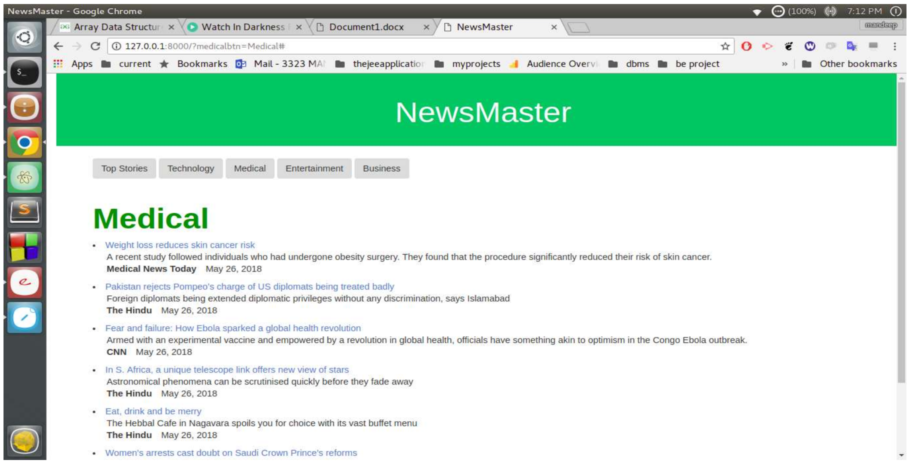
The home url of NewsMaster is set to Top Stories, which shows headlines across the world and you can access it using http://127.0.0.1:8000/ from your favourite web browser.



Technology

- 5 ways deception tech is disrupting cybersecurity
  Enterprises and their Security Operations Centers (SOCs) are under siege. Security events are being triggered from all corners of the security stack – from the firewall, endpoints, and ...
  The Next Web May 26, 2018
- How a 28-year-old went from running poker games and waitressing on roller skates to running her own 'InsurTech' startup
  Phoebe Hugh, 28, runs startup Brolly, an artificial intelligence-powered chatbot that helps people navigate the world of insurance. She is featured on Business Insider's UK Fintech 35 under 35. Hugh worked as an Indian head masseuse, roller-skating waitress, ...
  Business Insider May 26, 2018
- Lights, camera, adventure!
  An eventful road trip, an ambitious journey, a cross-country chase... experience a thrilling getaway from the comfort of your home.
  The Hindu May 26, 2018
- South Korea relieved about Trump-Kim summit revival efforts
  South Korea on Saturday expressed cautious relief about the revived talks for a summit between President Donald Trump and North Korean leader Kim Jong Un.
  CNBC May 26, 2018
- Fitness contest turns into 'fuel challenge' - The Hindu
  The Hindu Fitness contest turns into 'fuel challenge' The Hindu In the fraught ecosystem of social media, it would appear that even something as innocuous as a motivational fitness video is not immune to the daily cut-and-thrust between political adversaries

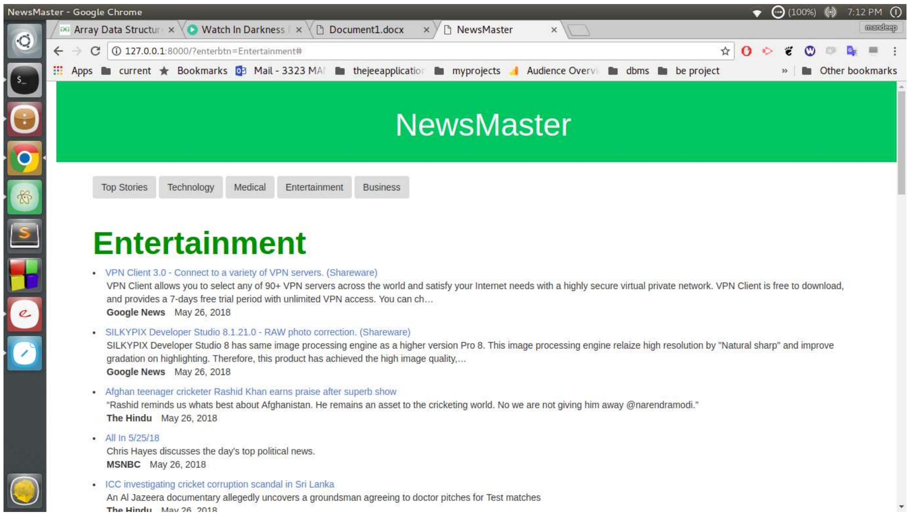
Fig.10.4 Technology news



Medical

- Weight loss reduces skin cancer risk
  A recent study followed individuals who had undergone obesity surgery. They found that the procedure significantly reduced their risk of skin cancer.
  Medical News Today May 26, 2018
- Pakistan rejects Pompeo's charge of US diplomats being treated badly
  Foreign diplomats being extended diplomatic privileges without any discrimination, says Islamabad
  The Hindu May 26, 2018
- Fear and failure: How Ebola sparked a global health revolution
  Armed with an experimental vaccine and empowered by a revolution in global health, officials have something akin to optimism in the Congo Ebola outbreak.
  CNN May 26, 2018
- In S. Africa, a unique telescope link offers new view of stars
  Astronomical phenomena can be scrutinised quickly before they fade away
  The Hindu May 26, 2018
- Eat, drink and be merry
  The Hebbal Cafe in Nagavara spoils you for choice with its vast buffet menu
  The Hindu May 26, 2018
- Women's arrests cast doubt on Saudi Crown Prince's reforms

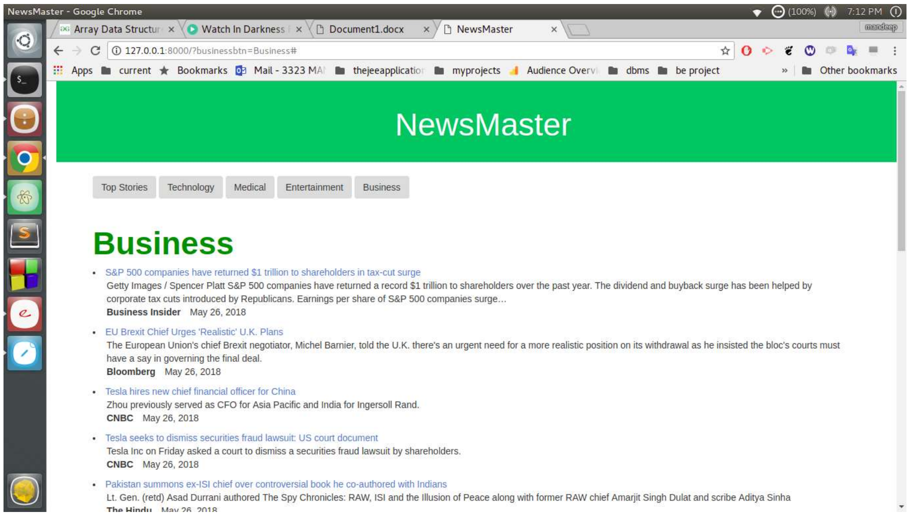
Fig.10.5 Medical News



Entertainment

- VPN Client 3.0 - Connect to a variety of VPN servers. (Shareware)
  VPN Client allows you to select any of 90+ VPN servers across the world and satisfy your Internet needs with a highly secure virtual private network. VPN Client is free to download, and provides a 7-days free trial period with unlimited VPN access. You can ch…
  Google News May 26, 2018
- SILKYPIX Developer Studio 8.1.21.0 - RAW photo correction. (Shareware)
  SILKYPIX Developer Studio 8 has same image processing engine as a higher version Pro 8. This image processing engine relaize high resolution by "Natural sharp" and improve gradation on highlighting. Therefore, this product has achieved the high image quality,…
  Google News May 26, 2018
- Afghan teenager cricketer Rashid Khan earns praise after superb show
  "Rashid reminds us whats best about Afghanistan. He remains an asset to the cricketing world. No we are not giving him away @narendramodi."
  The Hindu May 26, 2018
- All In 5/25/18
  Chris Hayes discusses the day's top political news.
  MSNBC May 26, 2018
- ICC investigating cricket corruption scandal in Sri Lanka
  An Al Jazeera documentary allegedly uncovers a groundsman agreeing to doctor pitches for Test matches
  The Hindu May 26, 2018

Fig. 10.6 Entertainment News



Business

- S&P 500 companies have returned $1 trillion to shareholders in tax-cut surge
  Getty Images / Spencer Platt S&P 500 companies have returned a record $1 trillion to shareholders over the past year. The dividend and buyback surge has been helped by corporate tax cuts introduced by Republicans. Earnings per share of S&P 500 companies surge…
  Business Insider May 26, 2018
- EU Brexit Chief Urges 'Realistic' U.K. Plans
  The European Union's chief Brexit negotiator, Michel Barnier, told the U.K. there's an urgent need for a more realistic position on its withdrawal as he insisted the bloc's courts must have a say in governing the final deal.
  Bloomberg May 26, 2018
- Tesla hires new chief financial officer for China
  Zhou previously served as CFO for Asia Pacific and India for Ingersoll Rand.
  CNBC May 26, 2018
- Tesla seeks to dismiss securities fraud lawsuit: US court document
  Tesla Inc on Friday asked a court to dismiss a securities fraud lawsuit by shareholders.
  CNBC May 26, 2018
- Pakistan summons ex-ISI chief over controversial book he co-authored with Indians
  Lt. Gen. (retd) Asad Durrani authored The Spy Chronicles: RAW, ISI and the Illusion of Peace along with former RAW chief Amarjit Singh Dulat and scribe Aditya Sinha
  The Hindu May 26, 2018

Fig. 10.7 Business News

CHAPTER 11

DEPLOYMENT AND MAINTENANCE

INSTALLATION STEPS:

There are many software which are important to our project. Our project is completed developed in python-Django framework virtual environment, so any user doesn't have install all the dependencies, all the dependencies are already installed in the project environment.

Installing the virtual environment (project environment name: vnews)

```
python3 –m venv vnews
```

Activate the virtual environment using:

```
source vnews/bin/activate
```

For fetching news online we used https://newsapi.org/v2/, you need to get the API-KEY to use newsapi. You can simply get API-KEY by registering on the URL https://newsapi.org/v2/ using email-id and use that API-KEY in the code.

Set the API-KEY in newsapp/views.py.

Since all the dependencies are all set up in the virtual environment, you don't have worry about installing any of the dependencies, just start the Django server which is set up on localhost, you can change the server URL settings in settings.py

```
python manage.py runserver
```

The superuser credentials of our project are, you can reset them by using the below credentials:

Username: mandeep

Email: msdeep14.ms@gmail.com

Password: pawan231

UNINSTALLATION:

For uninstalling the softwares installed, you can just remove the set-up virtual environment.

CHAPTER 12

CONCLUSION AND FUTURE WORK

CONCLUSION

In this project, we improve the quality of news cache and recommendations by predicting popularity of articles based on their content prior to publishing. The need for predicting popularity of article arises from the stiff competition among different news agencies and aggregators and enormous content available on internet due to which most useful content hide and some unusual news appears on top.

We predict the most popular pieces in different categories to provide the set of most popular news headlines, which is then can be used for multiple use-cases like in content caching, advertising, forecasting and recommendation.

FUTURE WORK

- Though content is main identifier for extracting features, but some fraction can be given to online popularity of news, like number of shares on Facebook or number of retweets on twitter or number of clicks.
- In our current project, we have global content relevancy algorithm, it can be further developed for specific user profiles.

REFERENCES

[1] Martin Weber and Maarten H. Lammers "Finding news in Haystack - Event based news clustering with social media based ranking"

[2] Roja Bandari, SitaRam Asur, Bernando A. Huberman "The Pulse of News in Social Media: Forecasting Popularity"

[3] Xianshu Zhu and Tim Oates "Finding story chains in newswire articles using random walks"

[4] Xiang Xang, Junbo Zhao, Yann LeCun "Character- level Convolutional Networks for Text Classification"

[5] Lewis, D. D. Naive (Bayes) at forty: The independence assumption in information retrieval. MachineLearning: ECML-98, Tenth European Conference on Machine Learning 1998

[6] Kecman V. "Learning and Soft Computing, Support Vector machines, Neural Networks and Fuzzy Logic Models," The MIT Press, Cambridge, MA, 2001.

[7] Using TF-IDF to Determine Word Relevance in Document Queries

[8] http://scikit-learn.org/dev/_downloads/scikit-learn-docs.pdf

[9] https://radimrehurek.com/gensim/models/doc2vec.html

[10] https://textblob.readthedocs.io/en/dev/

[11] https://tartarus.org/martin/PorterStemmer/

[12] https://docs.oracle.com/cd/B28359_01/text.111/b28304/astopsup.htm#CCREF1400

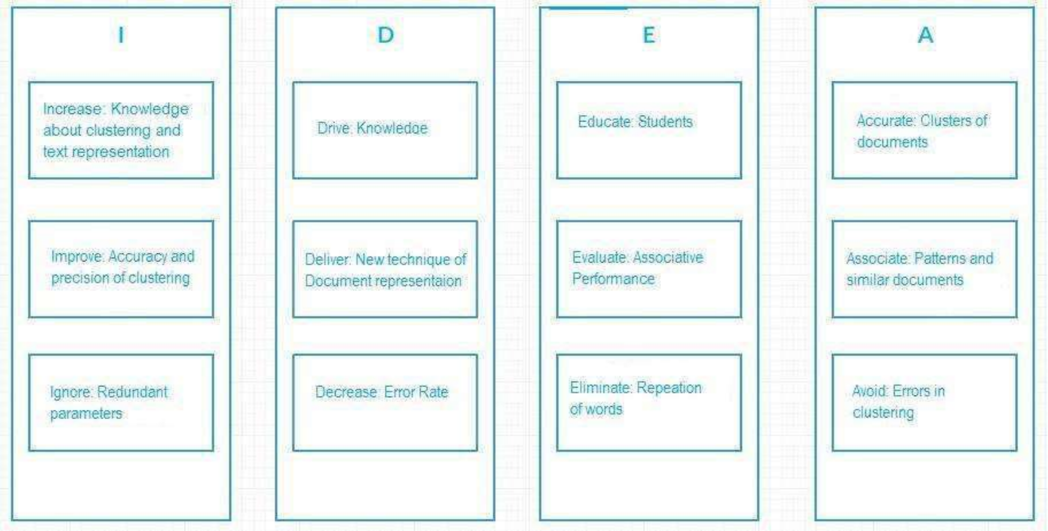
[13] http://www.vision.caltech.edu/anelia/DataPruning/

[14] http://www.kdnuggets.com/2015/03/deep-learning-text-understanding-from-scratch.html

ANNEXURE A

LABORATORY ASSIGNMENTS ON PROJECT ANALYSIS OF ALGORITHMIC DESIGN

To develop the problem under consideration and justification of feasibility using concepts of knowledge canvas and IDEA Matrix.



I	D	E	A
Increase: Knowledge about clustering and text representation	Drive: Knowledge	Educate: Students	Accurate: Clusters of documents
Improve: Accuracy and precision of clustering	Deliver: New technique of Document representaion	Evaluate: Associative Performance	Associate: Patterns and similar documents
Ignore: Redundant parameters	Decrease: Error Rate	Eliminate: Repeation of words	Avoid: Errors in clustering

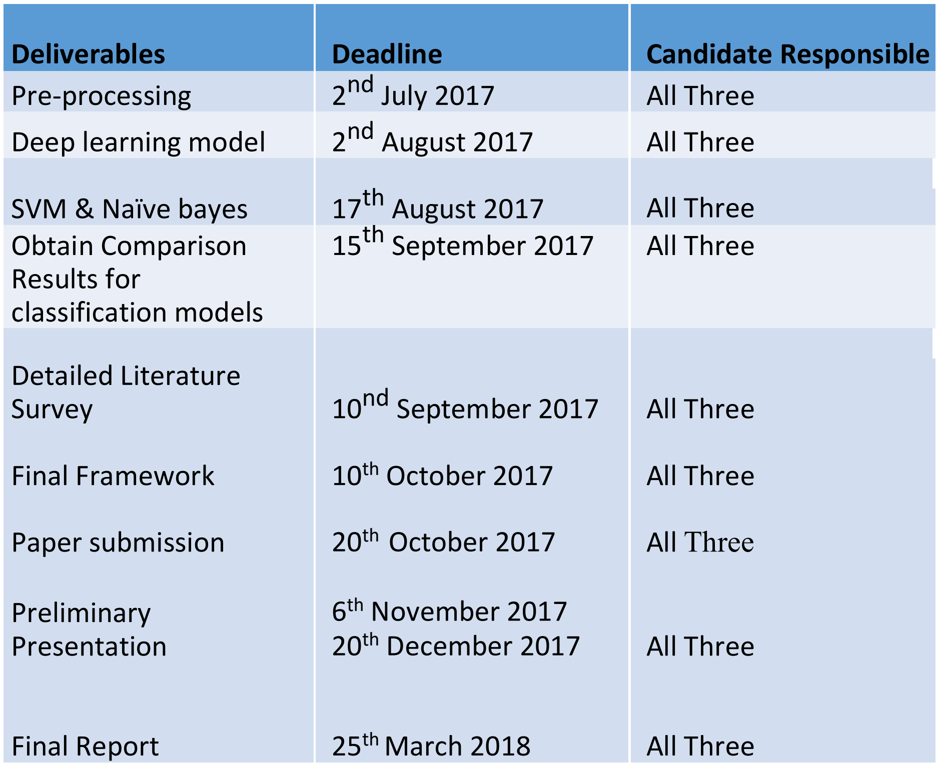
Figure A.1: IDEA Matrix

- This problem is an NP-Complete problem as representation of all news headlines and then applying character level convolution networks or naïve bayes classifier or SVM classifier will require large time which is not possible to define in polynomial time.

ANNEXURE B

PROJECT PLANNER

PROJECT PLAN



Deliverables	Deadline	Candidate Responsible
Pre-processing	2nd July 2017	All Three
Deep learning model	2nd August 2017	All Three
SVM & Naïve bayes	17th August 2017	All Three
Obtain Comparison Results for classification models	15th September 2017	All Three
Detailed Literature Survey	10nd September 2017	All Three
Final Framework	10th October 2017	All Three
Paper submission	20th October 2017	All Three
Preliminary Presentation	6th November 2017 20th December 2017	All Three
Final Report	25th March 2018	All Three

Figure C.1: Timeline Chart

Appendix C

Reviewers Comments of Paper Submitted

1. Paper Title:

   " Extracting News from Online Database – News Clustering
   based on Content Ranking"

2. Name of the Conference/Journal where paper submitted:
   - International Journal of Innovations & Advancement in Computer Science
   - International Journal of Computer Science
   - International Conference on Inventive Computing Systems and Applications
   - International journal of Latest Technology in Engineering, Management, and Applied Science
3. Paper accepted/rejected:

   Accepted in all conferences, mentioned above.

4. Review comments by reviewer
   - You are expected to redraft your article technically as now it is based on application.
   - Increase the number of references.

Appendix D

Plagiarism Report

ORIGINALITY REPORT

18%	22%	18%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

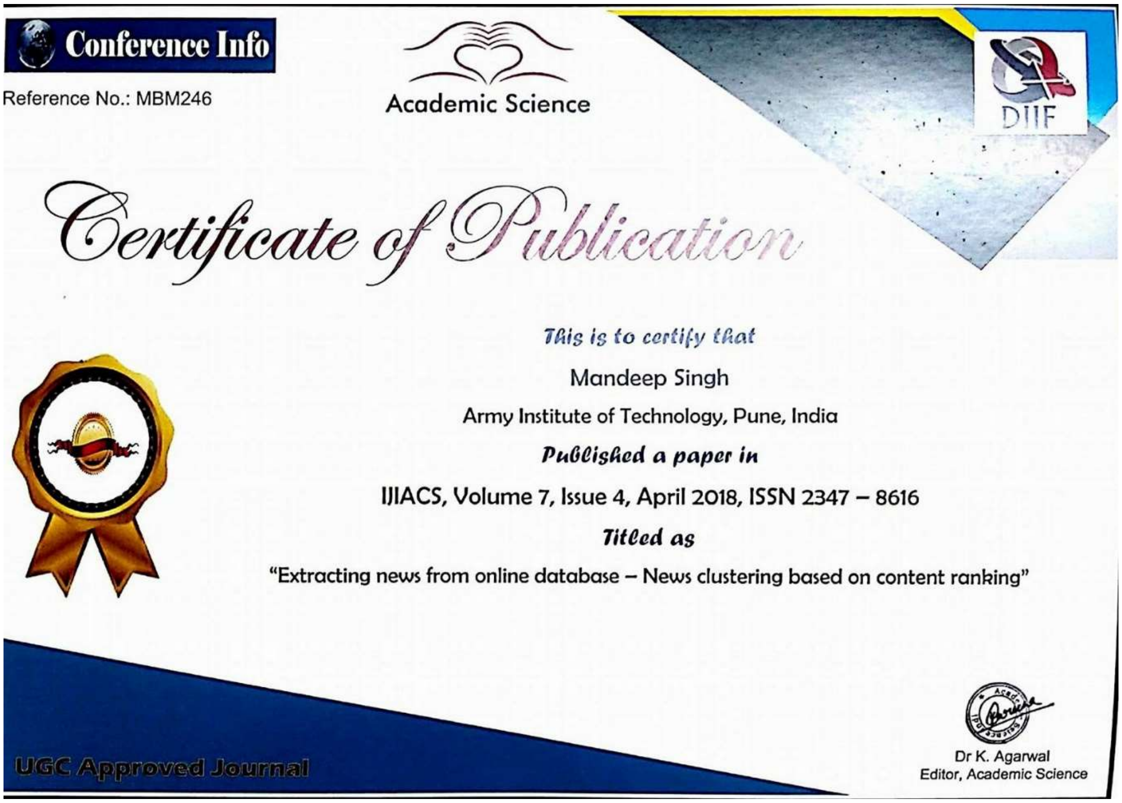
PRIMARY SOURCES

1	mediatechnology.leiden.edu Internet Source	7%
2	arxiv.org Internet Source	3%
3	Martin Weber, Maarten H. Lamers. "Finding News in a Haystack - Event Based News Clustering with Social Media Based Ranking", 2013 International Conference on Cloud and Green Computing, 2013 Publication	3%
4	en.wikipedia.org Internet Source	2%
5	docplayer.net Internet Source	1%
6	www.scribd.com Internet Source	1%
7	orbilu.uni.lu Internet Source	1%

Appendix E

Information of Project Group Members

Name	- Mandeep Singh
Date of Birth	- 14 August 1995
Gender	- Male
Permanent Address	- Sikar, Rajasthan
E-mail	- mandeepsingh_14070@aitpune.edu.in
Mobile/Contact No.	- 7769942097
Paper Published	-“ Extracting News from Online Database – News Clustering based on content Ranking”



Conference Info

Reference No.: MBM246

Academic Science

DIIF

Certificate of Publication

This is to certify that

Mandeep Singh

Army Institute of Technology, Pune, India

Published a paper in

IJIACS, Volume 7, Issue 4, April 2018, ISSN 2347 – 8616

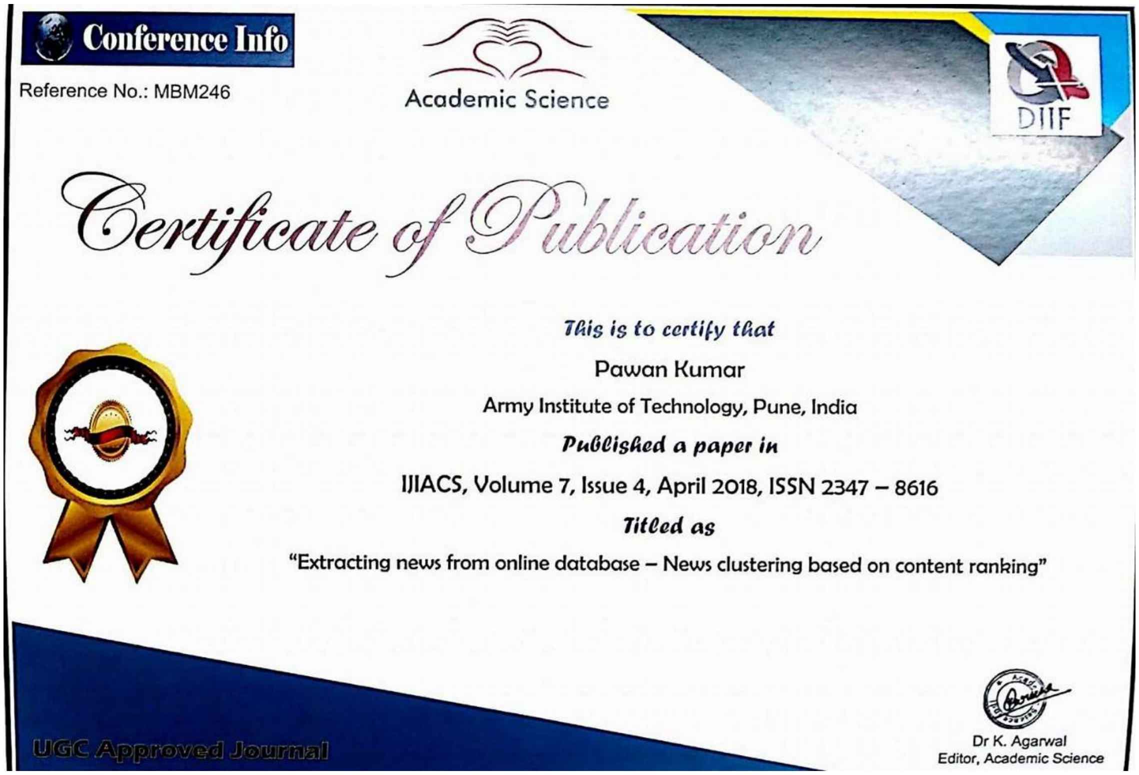
Titled as

“Extracting news from online database – News clustering based on content ranking”

UGC Approved Journal

Dr K. Agarwal
Editor, Academic Science

Name	- Pawan Kumar
Date of Birth	- 20 January 1995
Gender	- Male
Permanent Address	- Bhiwani, Haryana
E-mail	- pawankumar_14081@aitpune.edu.in
Mobile/Contact No.	- 7030305512
Paper Published	- “Extracting News from Online Database – News Clustering based on content Ranking”



Conference Info

Reference No.: MBM246

Academic Science

DIIF

Certificate of Publication

This is to certify that

Pawan Kumar

Army Institute of Technology, Pune, India

Published a paper in

IJIACS, Volume 7, Issue 4, April 2018, ISSN 2347 – 8616

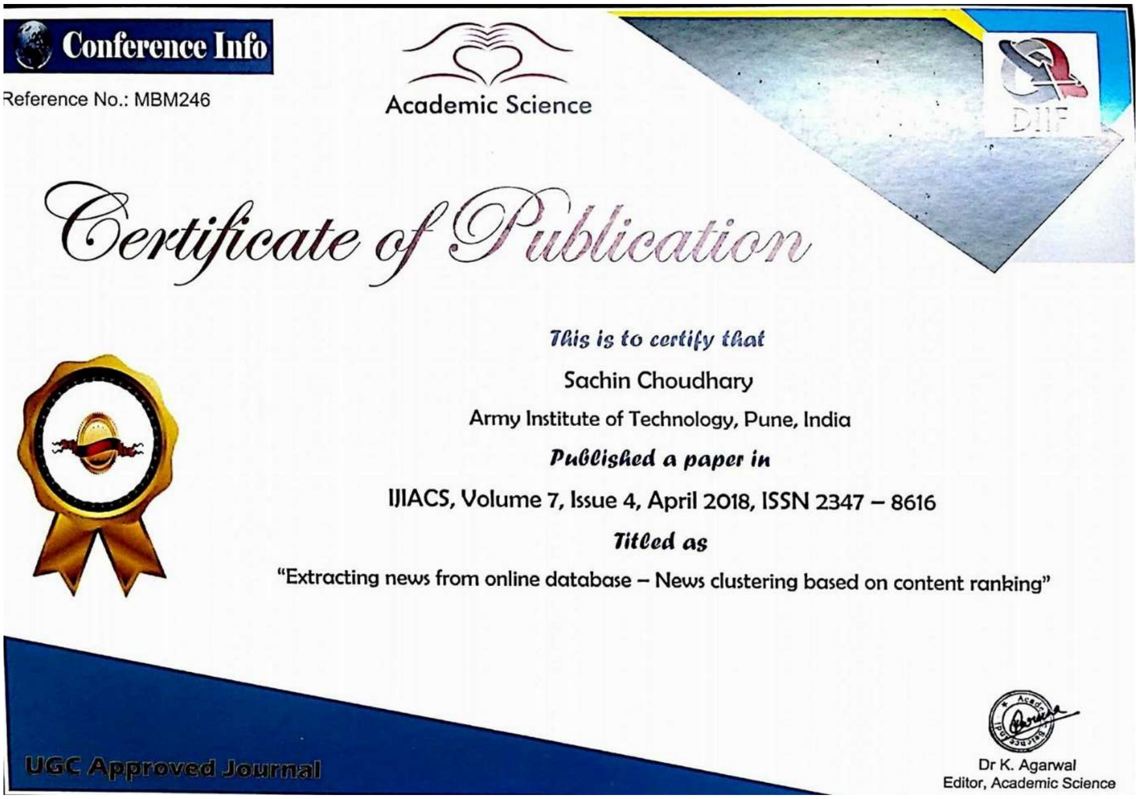
Titled as

“Extracting news from online database – News clustering based on content ranking”

UGC Approved Journal

Dr K. Agarwal
Editor, Academic Science

Name - Sachin Choudhary
Date of Birth - 12 September 1995
Gender - Male
Permanent Address - Jhunjhunu, Rajasthan
E-mail - sachinchoudhary_14276@aitpune.edu.in
Mobile/Contact No. - 9764002579
Paper Published - "Extracting News from Online Database – News Clustering based on content Ranking"



Conference Info

Reference No.: MBM246

Academic Science

DIIF

Certificate of Publication

This is to certify that

Sachin Choudhary

Army Institute of Technology, Pune, India

Published a paper in

IJIACS, Volume 7, Issue 4, April 2018, ISSN 2347 – 8616

Titled as

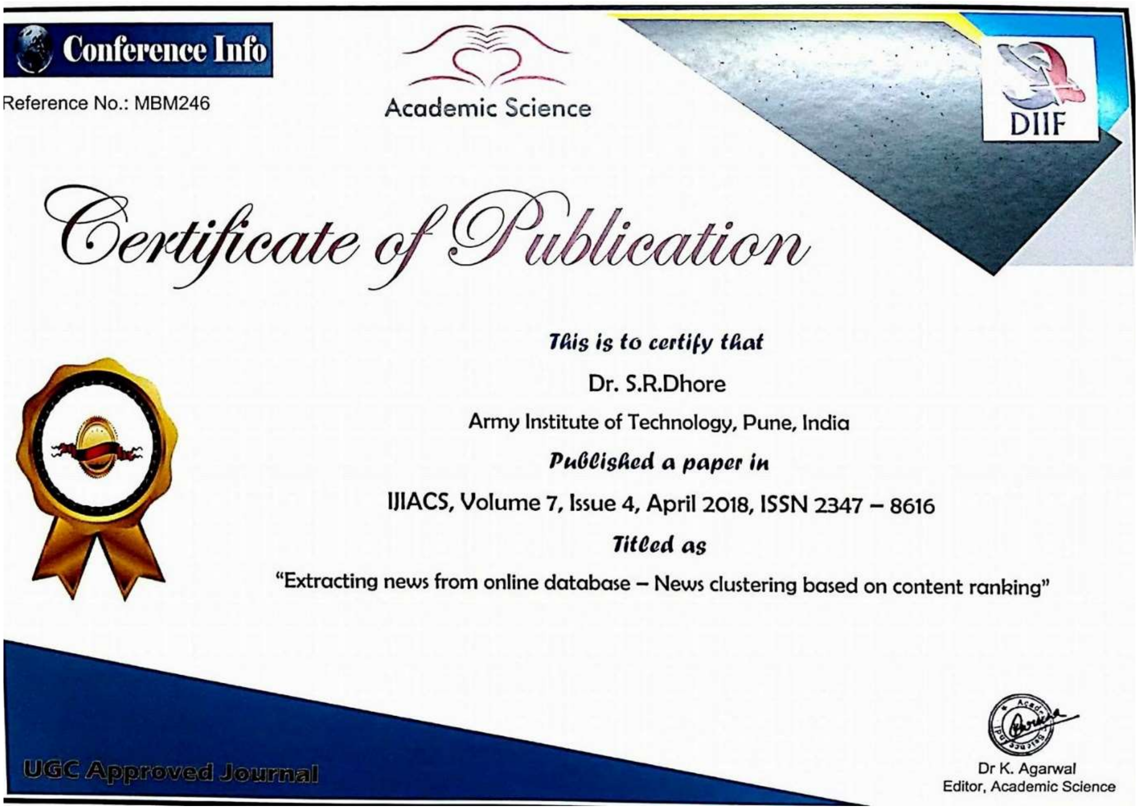
"Extracting news from online database – News clustering based on content ranking"

UGC Approved Journal

Dr K. Agarwal
Editor, Academic Science

Project Guide

Name - Prof (Dr) S.R. Dhore
E-mail - hodcomp@aitpune.edu.in
Mobile/Contact No. - 9890809251
Paper Published - “Extracting News from Online Database
– News Clustering based on content Ranking”



Conference Info

Reference No.: MBM246

Academic Science

DIIF

Certificate of Publication

This is to certify that

Dr. S.R.Dhore

Army Institute of Technology, Pune, India

Published a paper in

IJIACS, Volume 7, Issue 4, April 2018, ISSN 2347 – 8616

Titled as

“Extracting news from online database – News clustering based on content ranking”

UGC Approved Journal

Dr K. Agarwal
Editor, Academic Science