**News Article Analysis**

**Table

Description automatically generated**

**Submitted to  
 Prof. Geraldine Bessie Amali, SCOPE**

**School of Computer Science and Engineering**



**Contents**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Topic** | **Page No** |
| 1. | Abstract |  |
| 2. | Introduction |  |
| 3. | Hardware / Software Requirements |  |
| 4. | Existing System/approach/method |  |
| 5. | Drawback/limitations of existing System/approach/method |  |
| 6. | Proposed/Developed Model |  |
| 7. | Design |  |
| 8. | Module Wise Description |  |
| 9. | Implementation |  |
| 10. | Results and Discussion |  |
| 11. | Conclusion |  |
| 12. | Sample Code |  |

**Abstract**

News Article Analysis is a very important application of Natural Language Processing (NLP). Whenever there is a publish of a news article, happening it involves various types of discussions. Important information always gets lost while we are reading and listening news from various sources such as article, podcasts and reading various newspaper and magazines. Consequently, it becomes extremely essential to extract those important key points for future reference. In this project we aim to implement the analysis of the news articles . There are three techniques used here – news translation, news categorization and news summarization. We have used both traditional machine learning model and Deep learning techniques.

1. **Introduction**

Our project mainly focuses on text translator, summarizer and categorizer and text is based upon news.

In our day-to-day news story, recent or current events are covered, either   
generally (as in daily newspapers) or in relation to a particular subject (i.e. political or trade news magazines, club newsletters, or technology news websites). Reports from people who witnessed an occurrence can be included in news articles. When we visit a news website, we must have seen that the news is separated into categories. Tech, entertainment, sports, and other popular categories can be found on practically all news websites. This post is for you if you want to learn how to categorize news categories using machine learning.

Before releasing an item, every news website organizes it into different categories so that users may quickly select the categories of news that interest them. For instance, I frequently visit news websites and click on the technology section because I want to read about the most recent technological developments. The readers might be not knowing the news category, so they can enter the news and can know the category.

The process of creating a concise, fluid, and, most importantly, accurate summary of a lengthy text content is known as text summarization. The fundamental goal of automatic text summarization is to be able to extract the most important information from a large body of text and display it in a way that is human readable. Automatic text summarizing techniques could be particularly beneficial as online textual data increases since more informative material can be viewed quickly.

1. **Hardware / Software Requirement**

🡪 Python 3.9 with GPU integrated

🡪 Flask (Python Back-end framework)

🡪 Python library : Numpy, Pandas, Scikit-learn, Tensor-flow  
🡪 Visualization tools : Matplotlib, Seaborn  
🡪 Data Preprocessing tools : Spacy

1. **Existing System / approach method**

**3.1) Drawback / Limitations of existing System/approach/method**

**Text translator** 🡪 Rule Based Machine Translation with Statistical method

**Text classifier** 🡪 Multiclass-classification

**Text Summarizer** 🡪 Extractive Summarizer  
 🡪 Abstractive Summarizer

There are many automated approaches for the above mentioned process. Since doing manual text pre-processing, vector generation and model fitting are tedious procedures we are opting Spacy framework which will make our initial pre-processing phase in an easy manner.

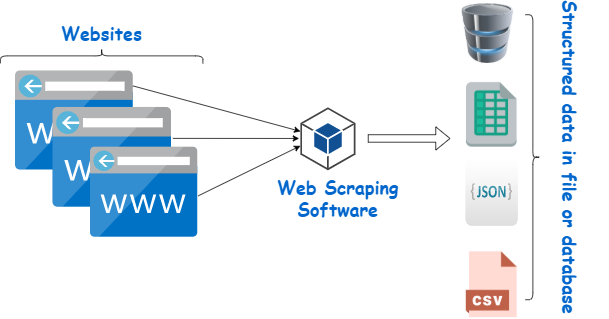
1. **Proposed / Developed Model**

We have optimized our LSTM model by modifying the regularization parameters with specific to our data-set. Since all the modules are implemented well-verse for generalized data-set we our aiming to integrated all these features under a single roof for user-convenience.

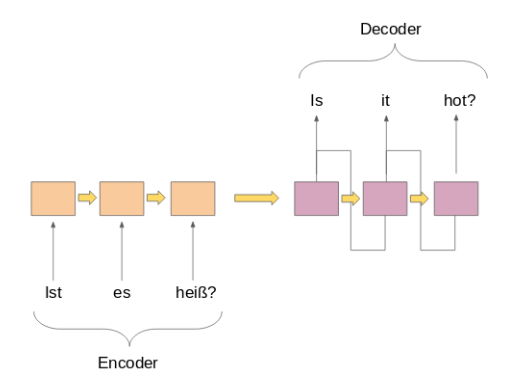
So we picked up News domain, since now-a-days people are not having enough time to read / listen the entire news content we are providing the top-most and demanded feature which our above mentioned functionality.

**4.1) Design**

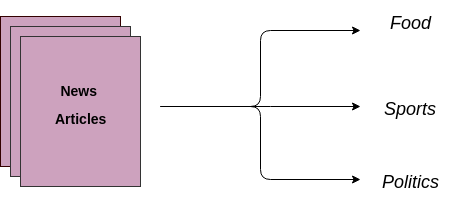
**Module-1 : Web Scrapping**



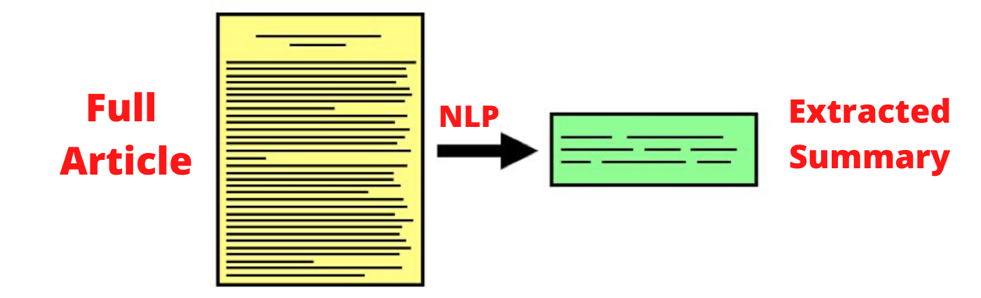
**Module-2 : Text translator**



**Module-3 : Text classifier**



**Module-4 : Text Summarizer**



**4.2) Module Wise Description**

**Module-1 : Web Scrapping**

Web scraping is a computerised technique for gathering copious volumes of data from websites. The majority of this data is unstructured in HTML format and is transformed into structured data in a database or spreadsheet so that it can be used in multiple applications. To collect data from websites, web scraping can be done in a variety of methods. These include leveraging specific APIs, online services, or even writing your own code from scratch for web scraping.

**Module-2 : Text translator** 🡪 Source language Identification using Multinomial Naïve bayes Algorithm 🡪 Identified language Translation using Sequence-to-Sequence model

**Module-3 : Text classifier** 🡪 Traditional Machine Learning and LSTM model.

**Module-4 : Text Summarizer** 🡪 Abstractive Text summarizer using Sequence-to-Sequence LSTM model

**4.3) Implementation**

**Module-1 : Web Scrapping**

**Module-2 : Text translator**

**Module-3 : Text classifier**

**Module-4 : Text Summarizer**

**5) Results and Discussions**

**6) Conclusion (Mention limitations in your project and how it can be enhanced)** The major drawbacks of the our implementation (especially LSTM model) is lackin accuracy and biased towards the data-set. Though we tried to bring the accuracy score near to the un-biased solution we can’t able to achieve. So our aim is to improve the accuracy measure so that our model is generalized and capture quite a lot of semantics into the domain.

**References**

**Appendix**

<https://towardsdatascience.com/how-to-detect-and-translate-languages-for-nlp-project-dfd52af0c3b5>