
# TRIBHUVAN UNIVERSITY

## INSTITUTE OF ENGINEERING

## THAPATHALI CAMPUS

**PROJECT NO.: THA079MSISE014**

# NLP AND WEB SCRAPING FOR NEPALI AGRICULTURAL DATA ANALYSIS

**BY**

**SAMEER GAUTAM**

**A PROJECT**

**SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATICS AND INTELLIGENT SYSTEMS ENGINEERING**

**DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING**

**KATHMANDU, NEPAL**

**AUGUST, 2024**

# NLP and Web Scraping for Nepali Agricultural Data Analysis

by

Sameer Gautam

THA079MSISE014

Project Supervisor

Asst. Professor Praches Acharya

A project submitted in partial fulfillment of the requirements for the degree of

Master of Science in Informatics and Intelligent Systems Engineering

Department of Electronics and Computer Engineering

Institute of Engineering, Thapathali Campus

Tribhuvan University

Kathmandu, Nepal

August, 2024

## ACKNOWLEDGMENT

The successful completion of this project would not have been possible without the invaluable contributions and support of numerous individuals.

First and foremost, I extend my deepest gratitude to my supervisor, **Asst. Professor Praches Acharya**, of **TCIOE**, for his exceptional guidance, insightful feedback, and unwavering encouragement throughout the course of this research. His expertise and mentorship have been instrumental in shaping this project. I would also like to express my sincere appreciation to the M.Sc. coordinator, **Asst. Professor Dinesh Baniya Kshatri**, for his efficient coordination, astute critiques, and unwavering patience in overseeing the project work.

I am deeply indebted to my classmates and friends for their invaluable advice, unwavering support, and camaraderie, which have been a constant source of motivation and encouragement.

To my family, I am eternally grateful for their unconditional love, encouragement, and unwavering belief in my abilities. Their support has been the bedrock upon which I have built my academic pursuits and personal aspirations. I am especially thankful to my parents for their emotional support, unwavering faith in my abilities, and constant encouragement to pursue my dreams.

I would also like to acknowledge the assistance and cooperation of all those who have contributed to this project, whether directly or indirectly. Your collective efforts have been invaluable in bringing this research to fruition.

**Sameer Gautam**
THA079MSISE014
August, 2024

## ABSTRACT

Nepalese farmers face challenges accessing vital agricultural information due to language barriers and the dominance of the Nepali language in rural areas. This research proposes utilizing social media data and Natural Language Processing (NLP) techniques to analyze Nepali language content and gain insights into the specific needs and challenges of these farmers. By understanding their priorities and concerns, agricultural extension services can tailor interventions, improve communication, and ultimately empower Nepalese farmers with accessible and relevant agricultural knowledge. This project will focus on developing effective methods for collecting and analyzing Nepali social media data, training NLP models to accurately interpret agricultural content in Nepali, and creating user-friendly tools to disseminate the findings to both farmers and extension agents. The anticipated outcomes include enhanced communication, data-driven decision-making, and the overall empowerment of Nepalese farmers through improved access to agricultural information.



**Keywords:** Agricultural knowledge, communication, empowerment, farmer needs, language barriers, Nepalese agriculture, Natural Language Processing (NLP), social media.

## TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## LIST OF ABBREVIATIONS

| | |
|---|---|
| API | Application Programming Interface |
| BERT | Bidirectional Encoder Representations from Transformers |
| DL | Deep Learning |
| ML | Machine Learning |
| NL | Natural Language |
| NLIDBs | Natural Language Interfaces to Databases |
| NLP | Natural Language Processing |
| NLTK | Natural Language Toolkit |
| RNN | Recurrent Neural Network |
| TF-IDF | Term Frequency - Inverse Document Frequency |
| LDA | Latent Dirichlet Allocation |

# 1 INTRODUCTION

## 1.1 Background

Nepal's agricultural sector, the lifeblood of many citizens, grapples with modernization and sustainability. Communication barriers and limited access to modern practices, especially in rural areas where Nepali language reign supreme, restrict the flow of vital agricultural knowledge.

Communication between farmers and extension services is a hurdle in agricultural development. In rural regions, local languages prevail and traditional extension services struggle to effectively distribute crucial information on improved farming practices, market trends and government agricultural policies.

Digital technologies, such as social media platforms, have opened new avenues for addressing communication challenges in agriculture. Lack of digital literacy in rural communities and linguistic diversity are obstacles that need to be overcome to benefit Nepalese farmers.

The project aims to use social media data to bridge the communication gap and empower Nepalese farmers. The project seeks to uncover valuable insights into farmer needs, challenges and priorities by developing methods to analyze social media content in local languages. Through this approach, agricultural extension services can better tailor their interventions to address the specific requirements of diverse farming communities, thus contributing to the sustainable development of Nepal's agricultural sector.

## 1.2 Motivation

Digital technologies, particularly social media, offer potential solutions to agricultural challenges. However, current approaches often neglect Nepal's linguistic diversity, hindering effective communication with farmers. This project aims to overcome this by utilizing social media data and developing advanced Natural Language Processing (NLP) tools tailored to Nepali languages. By providing timely, relevant agricultural information in local dialects, we seek to empower farmers, enhance productivity, improve livelihoods, and build resilience within Nepal's agricultural sector.

## 1.3 Problem Definition

Nepalese agriculture faces a significant obstacle due to the lack of tools and methodologies specifically designed to understand and address the unique needs of farmers

who primarily communicate in the Nepali language. Existing approaches often overlook or fail to adequately account for this linguistic diversity, resulting in a disconnect between agricultural interventions and the actual realities faced by farmers on the ground. This language barrier hinders effective communication and knowledge sharing, preventing farmers from accessing crucial information and resources that could enhance their agricultural practices and livelihoods.

Furthermore, the dominance of the Nepali language in rural areas, where a majority of farmers reside, exacerbates this issue. Traditional agricultural extension services and information dissemination channels often rely on materials and communication methods that are not easily accessible or understandable for Nepali-speaking farmers. This linguistic divide not only limits farmers' access to knowledge but also restricts their ability to voice their concerns, share their experiences, and participate in agricultural decision-making processes.

### 1.4 Project Objectives

- Deliver actionable insights to Nepali farmers by analyzing social media data using NLP techniques.
- Overcome linguistic and technical challenges in processing Nepali social media content.

### 1.5 Scope of Project

This project aims to utilize social media data and Natural Language Processing (NLP) techniques to better understand the challenges and needs of Nepalese farmers. By analyzing Nepali language content on platforms such as Twitter and Facebook, we can extract valuable insights into the specific issues they face. The project will develop NLP models to analyze sentiment, identify key topics, and extract relevant information from posts made by farmers about their problems.

These insights will be employed to develop user-friendly tools and resources for government bodies and agricultural extension services. These tools will offer accessible agricultural information and facilitate the creation of tailored interventions to meet the specific needs of Nepalese farmers. By bridging the communication gap between

farmers and government entities, this project aims to empower both parties with valuable information and insights, promoting a more informed, responsive, and farmer-centric agricultural ecosystem in Nepal.

### 1.6 Potential Project Applications

The project's outcomes hold immense promise for various aspects of Nepalese agriculture, fostering a more informed and empowered agricultural sector. By harnessing social media data, the project can revolutionize how stakeholders tackle challenges and capitalize on opportunities.

- Precision Farming: Tailoring agricultural practices to specific local conditions is critical for maximizing yields and minimizing risks. Project insights can equip farmers with information on optimal planting times, suitable crop varieties, and effective pest and disease management strategies specific to their regions. This real-time, localized data empowers them to make informed decisions for increased productivity.

- Bridging the Market Gap: Accessing markets and staying informed about prevailing prices are vital for farmers to secure fair compensation for their produce. Social media analysis can shed light on market trends, demand fluctuations, and consumer preferences. This empowers farmers to strategize production and marketing efforts effectively, facilitating fairer and more efficient agricultural trade.

- Building Climate Resilience: Climate change poses a significant threat to agricultural productivity and food security. By analyzing social media data, the project can capture farmers' experiences related to changing weather patterns, crop failures, and adaptation strategies. This information can be used to develop climate-resilient farming practices and policies, ensuring farmers are better equipped to cope with climate challenges.

- Empowering Policy Decisions: Informed policymaking is crucial for addressing systemic challenges and fostering agricultural development. The project's insights into farmer needs, challenges, and priorities can inform the formulation of evidence-based policies and interventions tailored to the diverse needs of Nepalese

farming communities. By amplifying farmers' voices, the project contributes to more inclusive and effective agricultural governance.

In essence, the project's applications span the entire agricultural value chain, from production to marketing. It holds the potential to transform Nepalese agriculture, making it more resilient, sustainable, and prosperous. This project can be the reference for every other researchers who tries to help in agricultural sector by any means.

### 1.7 Originality of Project

This project aims to utilize social media data and Natural Language Processing (NLP) techniques to better understand the challenges and needs of Nepalese farmers. By analyzing Nepali language content on platforms such as Twitter and Facebook, we can extract valuable insights into the specific issues they face. The project will develop NLP models to analyze sentiment, identify key topics, and extract relevant information from posts made by farmers about their problems.

These insights will be employed to develop user-friendly tools and resources for government bodies and agricultural extension services. These tools will offer accessible agricultural information and facilitate the creation of tailored interventions to meet the specific needs of Nepalese farmers. By bridging the communication gap between farmers and government entities, this project aims to empower both parties with valuable information and insights, promoting a more informed, responsive, and farmer-centric agricultural ecosystem in Nepal.

### 1.8 Organisation of Project Report

This report is structured into four chapters to comprehensively delineate the project's objectives, methodologies, and anticipated outcomes.

Chapter 1 provides a foundational overview of the project, emphasizing its significance in addressing the unique challenges faced by Nepalese farmers. It clearly articulates the project's goals and potential impact on the agricultural sector.

Chapter 2 delves deeper into the theoretical underpinnings of the project, exploring the core Natural Language Processing (NLP) techniques that will be employed. It elucidates how these techniques will be adapted and refined to effectively analyze the complexities

of Nepali language and agricultural-related social media data.

Chapter 3 presents a detailed methodology for data acquisition, preprocessing, and the application of NLP algorithms. It outlines the specific steps involved in transforming raw social media data into actionable insights.

Chapter 4 showcases the expected outcomes of the project, including concrete examples of sentiment analysis, topic modeling, and named entity recognition results. It provides visualizations to aid understanding and employs rigorous evaluation metrics to assess the model's performance. Additionally, this chapter critically examines the project's strengths, limitations, and potential areas for future improvement.

## 2 LITERATURE REVIEW

”Bagheri et al. (2023) [1] explored the potential of social media text mining for agricultural insights, focusing on Oklahoma Panhandle farmers’ tweets about planting, harvesting, irrigation, and drought.

They used text mining to analyze Twitter data, aiming to assess its value for agricultural knowledge discovery and the suitability of existing sentiment analysis tools.

The study found that Twitter can effectively track the timing of key agricultural activities. However, it also revealed limitations in standard sentiment analysis for agricultural contexts, emphasizing the need for domain-specific tools.

The methodology involved collecting, cleaning, and analyzing farmers’ tweets, as well as evaluating sentiment analysis tools. The study’s strengths include its novel approach and data-driven findings, while limitations pertain to the need for specialized sentiment analysis and the study’s geographic focus.

Singh et al.(2022) [2] examined the impact of the COVID-19 pandemic on Indian agriculture by analyzing agricultural-related tweets across three phases of the Indian lockdown. Employing machine learning, they assessed sentiment and emotions expressed within these tweets.

Results indicated a prevalence of negative sentiment during the initial lockdown phase, suggesting widespread fear and uncertainty among agricultural stakeholders. However, this negativity subsided in subsequent phases. The study concludes that social media analytics can effectively monitor agricultural stakeholder sentiment during crises, informing policy responses.

The methodology involved collecting, cleaning, and analyzing tweets, utilizing machine learning for sentiment and emotion analysis, and comparing regional variations. While the study effectively leveraged social media data and machine learning, its scope is limited to India and Twitter, potentially excluding perspectives from other platforms or demographics. Additionally, a qualitative component could have enriched the understanding of underlying reasons for sentiment shifts.

Palaniswamy et al.(2022) [3] investigated the factors influencing the adoption of social media marketing (SMM) among agriculturists in South India. The study employed a structured questionnaire to collect data from 320 agriculturalists in Tamilnadu. The research model integrated the Theory of Planned Behavior (TPB) and the Technology Acceptance Model (TAM), incorporating factors influencing attitude such as perceived infotainment, perceived credibility, influence of reference groups, perceived usefulness, and ease of use.

The findings revealed that perceived credibility, reference group influence, perceived infotainment, and perceived usefulness significantly and positively impacted the adoption of SMM. However, perceived ease of use negatively affected the attitude towards SMM adoption. The study concludes that social media can be a valuable tool for agricultural marketing, and the identified factors play a crucial role in shaping agriculturists' attitudes toward its adoption.

The research contributes to the understanding of SMM adoption in the agricultural sector, particularly in the context of South India. The use of established theoretical frameworks (TPB and TAM) strengthens the study's validity. However, the study's limitations include its focus on a specific region and the potential for self-selection bias due to the convenience sampling method. Additionally, the study primarily focused on quantitative analysis, and incorporating qualitative insights could have provided a deeper understanding of the underlying motivations and barriers to SMM adoption among agriculturists.

Devienne et al.(2021) [4] explored the use of social media and Natural Language Processing (NLP) in natural hazard research. The study focused on automating the collection and classification of text data from ResearchGate, a social networking platform for researchers. The author employed web scraping using Selenium to gather a dataset of publications related to the keyword ”taenite.” Subsequently, the Word2Vec model was implemented to learn word embeddings and predict contexts based on single words. The model achieved an accuracy of approximately 78% in estimating contexts. The study further explored the classification of events like tsunamis and earthquakes based on word associations.

The research demonstrated the feasibility of using machine learning and NLP techniques to extract valuable information from social media data in the context of natural hazard research. The automated web scraping process using Selenium enabled the efficient collection of a large dataset from ResearchGate. The Word2Vec model proved effective in predicting contexts and identifying related terms, as evidenced by the visualization of nearest neighbors for the word ”earthquake” using Tensorboard. The study also explored the potential of classifying events based on word associations, although further refinement and validation are needed.

The strengths of this research lie in its innovative approach to utilizing social media data and NLP techniques for natural hazard research. The automated data collection and analysis methods offer efficiency and scalability. However, the study's limitations include the focus on a single social media platform (ResearchGate) and a specific keyword (”taenite”). The model's accuracy, while promising, could be further improved with larger and more diverse datasets. Additionally, the classification of events based on word associations requires further validation and refinement to ensure robustness and reliability.

In the paper ”Sentiment Analysis in Agriculture” [5] the authors Novak et al.(2021) proposed to examine the application of sentiment analysis in agriculture. The authors aimed to map the current state of research and suggest future directions. The review focused on scientific literature related to sentiment analysis in agriculture, excluding applications in related fields like food quality and production. The authors employed a conventional literature review methodology, searching Scopus and Google Scholar databases using specific keywords. The selected publications were analyzed and categorized based on their approaches, objects of analysis, and publication dates.

The review revealed a growing trend of research on sentiment analysis in agriculture, with machine learning being the most commonly used approach. The research primarily focused on analyzing the sentiment of farmers, the general public, and mainstream media towards agriculture. The authors concluded that sentiment analysis has potential applications in agriculture, particularly for analyzing public opinion and enhancing prediction models. However, they also highlighted the need for further research to improve the accuracy of ternary sentiment classification, which includes neutral sentiment.

The study’s strengths lie in its comprehensive overview of the existing literature and its identification of potential areas for future research. However, the limitations include the exclusion of research on sentiment analysis in related fields and the lack of in-depth analysis of individual studies. Additionally, the review could have benefited from a more critical evaluation of the methodologies and findings of the included publications.

# 3 METHODOLOGY

## 3.1 Theoretical Formulations

My project will leverage Natural Language Processing (NLP) techniques specifically tailored for the Nepali language to analyze social media data. The core of our approach involves:

**Basic Concept about the Chosen Model and Supporting Pre-/Post-Processing Steps:**

**Introduction to the Chosen Model:**

**Model Selection Rationale:**

The success of a data analysis project largely hinges on the selection of an appropriate model that aligns with the nature of the data and the objectives of the analysis. For this project, the chosen model is Naive Bayes for sentiment analysis or Latent Dirichlet Allocation (LDA) for topic modeling. The selection was based on the model's suitability for the specific characteristics of the dataset and its robustness in handling the type of analysis required.

**Model Overview:**

Naive Bayes is one of the best algorithm for sentiment analysis, particularly due to its simplicity and effectiveness.Naive Bayes is a probabilistic classifier based on Bayes' Theorem. It assumes that the features are independent given the class label, which is known as the naive assumption.

For the sentiment analysis aspect of the project, the Multinomial Naive Bayes classifier is commonly used. This is because the Multinomial Naive Bayes model is well-suited for text classification tasks, especially when the features are represented as word counts or term frequencies, which is typical in Natural Language Processing (NLP) applications. Why Multinomial Naive Bayes?

Text Classification: It works well for document classification tasks where the features are the frequencies of words or terms. Handling of Multiclass Problems: It can handle multiple classes, which is useful when the text data is categorized into multiple sentiment classes (e.g., positive, negative, neutral). Efficient and Fast: Multinomial Naive Bayes is computationally efficient, making it suitable for large datasets like those I might be working with in an agricultural data analysis context.

For models like Latent Dirichlet Allocation (LDA), the process involves discovering hidden topics within a collection of documents by assuming that each document is composed of a mixture of topics and each topic is composed of a mixture of words. In our project, which deals with sentiment analysis and topic modeling of Nepali agricultural data, LDA helps to reveal the underlying themes or topics that are present in the data. The model works by iteratively refining its estimates of how topics are distributed across documents and how words are distributed across topics, striving to best represent the observed data. This process allows us to categorize the content into meaningful topics, providing insights into common issues or themes among farmers, which can then be used to guide recommendations or inform policy decisions.

- Web Scraping: Web scraping is the automated technique of extracting data from websites using specialized software or scripts. This process enables the collection of large volumes of information from web pages, which can then be analyzed for various purposes. For our project, web scraping was crucial in gathering real-time data from social media platforms and agricultural forums where Nepali farmers discuss their experiences, challenges, and insights. Traditional data sources, like government reports or surveys, often fail to capture the timely and detailed perspectives needed to fully understand the daily issues faced by farmers. By scraping content from these online platforms, we were able to collect valuable, user-generated data that provided deeper insights into the agricultural practices and sentiments of Nepal's farming community. This data was indispensable for conducting sentiment analysis and topic modeling, allowing us to generate targeted suggestions that address the specific needs of these farmers.

- Pre-processing: Pre-processing in this project involves cleaning and preparing the raw Nepali text data from social media for subsequent analysis. The steps involved are:

  - Text Cleaning: This step involves removing noise and irrelevant information from the text data. This includes:

    * Removing emojis, URLs, and usernames, as these elements do not contribute to the understanding of the agricultural content.

    - Removing punctuation marks, as they are not necessary for the NLP analysis.
    - Removing hashtag values(#), as they are also not necessary for the NLP analysis.
    - Additionally, common Nepali stopwords are removed because they do not contribute significant meaning to the sentiment of a sentence.
  - Tokenization: This step involves splitting the cleaned Nepali text into individual words or meaningful units called tokens. This is crucial for further analysis as it allows the NLP models to process the text in a structured way. Tokenization in Nepali requires specific rules due to the unique characteristics of the language, such as the use of compound words and postpositions.

  By performing these pre-processing steps, the raw Nepali text data is transformed into a clean and standardized format that is suitable for further analysis using NLP techniques like sentiment analysis, topic modeling, and named entity recognition. This ensures that the subsequent analysis is accurate, relevant, and focused on the agricultural content of the social media posts.

- Sentiment Analysis: By converting text into numerical representations and training a machine learning model, sentiment analysis gauges public opinion within the data. Techniques like TF-IDF vectorization and algorithms such as Naive Bayes are employed to classify text as positive, negative, or neutral. Model performance is meticulously evaluated using metrics like accuracy, precision, recall, and F1-score. This process offers valuable insights into customer sentiment, brand reputation, and overall public perception.

  Following feature extraction, Naive Bayes, a machine learning model is trained on the numerical data. The model learns to associate specific word patterns with corresponding sentiments (positive or negative). Once trained, the model can predict the sentiment of new, unseen text data by assigning a sentiment label based on the learned patterns.

  The model's performance is then evaluated using metrics like accuracy, precision, recall, and F1 score. These metrics assess the model's ability to correctly classify sentiments. A confusion matrix can also be used to visualize the model's

performance by comparing the true sentiment labels with the predicted labels.

Sentiment analysis is a valuable tool in various applications. It helps businesses understand customer satisfaction, tracks public opinion on social media, informs market research strategies, and improves support systems by prioritizing issues based on emotional tone. Overall, sentiment analysis enables organizations to understand and respond effectively to the subjective opinions expressed in text data.

- Topic Modeling: To uncover latent themes within the text, topic modeling is applied. Algorithms like Latent Dirichlet Allocation (LDA) identify recurring patterns and group text into coherent topics. By analyzing the distribution of topics and their associated keywords, researchers can gain a deeper understanding of the primary concerns and interests of the target audience.

  Each identified topic will be represented by a set of keywords that are most strongly associated with it. By analyzing the prevalence and sentiment associated with different topics, the project can gain valuable insights into the key areas of interest and concern among Nepalese farmers. This information can then be used to inform the development of targeted interventions, educational resources, and policy recommendations that address the specific needs and challenges faced by the agricultural community in Nepal.

- Named Entity Recognition (NER): Named Entity Recognition (NER) is employed to identify and categorize specific entities like persons, organizations, locations, and dates within the text. A NER model is trained on Nepali text data using a pre-trained language model like BERT as a foundation. The model is fine-tuned on annotated data to accurately extract and classify named entities. Follow these fundamental steps:

  - Install Required Libraries: Begin by installing essential libraries such as transformers (for model handling), datasets (for data processing), torch (for deep learning functionalities), and matplotlib (for visualization).
  - Loading the Dataset: Nepali text data from an Excel file is loaded into a pandas DataFrame to manage.

- Load Pre-trained Model and Tokenizer: Access a pre-trained BERT model and tokenizer specifically designed for Nepali from the Hugging Face model hub. The tokenizer will segment the text into tokens, while the model serves as the base for NER fine-tuning.
- Annotate the Data: Employ the pre-trained NER model to annotate the Nepali text data with named entities. This involves tokenizing the text and aligning tokens with their respective entity labels to prepare for model training.
- Format Data for Model Input: Convert the annotated data into a format suitable for the Hugging Face datasets library. Ensure tokens and labels are correctly aligned and structured to meet the model's input requirements.
- Prepare Data for PyTorch: Transform the tokenized dataset into PyTorch tensors, essential for training deep learning models. Create data loaders to efficiently handle batch processing during model training.
- Define Optimizer and Scheduler: Define an optimizer (e.g., AdamW) to update the model's parameters during training. Implement a learning rate scheduler to adjust the learning rate dynamically, optimizing model performance.
- Train and Evaluate the Model: Train the NER model over multiple epochs. Compute the training loss, update the model weights accordingly, and assess its performance using validation data to ensure robustness and generalization.
- Visualize Training Progress: Utilize matplotlib to visualize and analyze the training and validation loss trends over epochs. This helps in monitoring model convergence, identifying potential issues like overfitting, and optimizing model training.

By following these systematic steps, you can effectively develop and evaluate a NER model tailored specifically for processing Nepali text data.

- Post-processing: Post-processing in this project refers to the refinement and summarization of the results obtained from the NLP analysis of Nepali social media data. The goal of post-processing is to derive meaningful insights and actionable recommendations from the raw output of the NLP models.

Two key post-processing techniques mentioned in the project are:

- Rule-based Filtering: This technique involves applying domain-specific rules to filter out irrelevant results from the NLP analysis. For example, after performing named entity recognition, the model might identify entities that are not relevant to agriculture, such as names of people or organizations. Rule-based filtering can be used to remove these irrelevant entities, leaving only those that are relevant to the agricultural context.
- Clustering: This technique aims to group similar topics or entities together based on their semantic similarity. For instance, after topic modeling, the model might identify several topics that are closely related, such as ”pest control” and ”disease management.” Clustering can be used to group these topics together, making it easier to interpret the results and identify overarching themes.

**Evaluation and Validation:** To measure the effectiveness of our model, we employ several key metrics:

- Accuracy, Precision, Recall, and F1-Score (for Classification Models): These metrics provide a comprehensive evaluation of the model's performance. Accuracy reflects the overall correctness of predictions. Precision and recall focus on how well the model performs with specific classes, while the F1-Score combines these metrics to provide a balanced measure of performance, especially important for datasets with class imbalances.
- Coherence Score (for Topic Models): This metric assesses the quality of the topics generated by the model. It evaluates whether the words grouped together in a topic are contextually relevant and whether the topics themselves are meaningful and easy to interpret. A higher coherence score indicates that the topics are more understandable and reflective of the data's underlying themes.

**Visualization:** Visual representations help to enhance the understanding of model results:

- Confusion Matrix (for Classification): This visualization shows the number of correct and incorrect predictions across different classes, helping to pinpoint areas where the model may be underperforming and providing insight into specific types of errors.
- Word Clouds and Topic Distributions (for Topic Modeling): Word clouds highlight the most frequent words within each topic, offering a visual summary of the key themes. Topic distributions reveal how topics are distributed across different documents, helping to understand the prevalence and relevance of each topic within the dataset.

By applying these post-processing techniques, the project can transform the raw output of the NLP models into a more organized and interpretable format. This can help to reveal hidden patterns, trends, and correlations in the data, leading to valuable insights into the needs and challenges faced by Nepalese farmers. These insights can then be used to inform the development of targeted interventions, educational resources, and policy recommendations that can benefit the agricultural community in Nepal.

**Major Benefits of the Chosen Technique:**

- Language Specificity: By focusing on Nepali language processing, we can accurately capture the nuances and sentiments expressed by Nepalese farmers in their native language.
- Targeted Insights: NLP techniques allow us to extract specific information relevant to agriculture, such as crop types, issues faced, and sentiment towards government policies.
- Actionable Recommendations: The analysis results can be used to inform targeted interventions, tailor extension services, and develop policies that address the specific needs of Nepalese farmers.

**Assumptions Taken into Account:**

- Data Availability: We assume the availability of sufficient social media data in the Nepali language related to agriculture.
- Data Quality: We assume that the collected data is representative of the broader Nepalese farming community and their concerns.
- NLP Resources: We assume the availability of appropriate NLP tools and resources for the Nepali language, including sentiment lexicons, labeled datasets, and pre-trained models.
- User Engagement: We assume that farmers and other stakeholders will actively participate in discussions on social media platforms, providing valuable data for analysis.

## 3.2 Mathematical Modelling

### 3.2.1 Pre-Processing

The initial phase involves text cleaning and normalization to prepare the Nepali text data for analysis. This includes:

1. **Tokenization:** Tokenization is the process of splitting the input text into smaller units called tokens (words, subwords, or characters). This step is crucial for both Sentiment Analysis and NER as it converts raw text into a structured format suitable for modeling.

$$\text{tokens} = \text{tokenizer}(\text{text}, \text{padding=True}, \text{truncation=True}) \tag{3.1}$$

2. **Removing Stopwords:**
   - **Punctuation Removal:**

$$T' = T - P \tag{3.2}$$

     where $T'$ is the cleaned text, $T$ is the original text, and $P$ is the set of punctuation marks.
   - **Stopword Removal:** $T' = T - S$, where $S$ are the stopwords or common words that may not carry significant meaning and are often removed during pre-processing.

3. **Text Normalization:**

    - **Unicode Normalization:**

$$T' = \text{normalize}(T) \tag{3.3}$$

    - **Diacritic Removal (if necessary):** Removing optional diacritical marks.
    - **Standardizing Variations:** Converting variations of words to a standard form.
    - **Removing HashTags:** In this hashtag value are removed for easy processig.

### 3.2.2 Sentiment Analysis:

Sentiment analysis involves determining the sentiment or emotion expressed in a piece of text, typically classifying it as positive, negative, or neutral. Here are the key mathematical concepts involved:

1. **Tokenization:** Tokenization is the process of splitting the input text into smaller units called tokens (words, subwords, or characters). Tokenizers encode text into input IDs, attention masks, and token type IDs.
2. **TF-IDF (Term Frequency-Inverse Document Frequency):** This is a statistical measure used to evaluate the importance of a word in a document relative to a collection of documents (corpus).

    - **Term Frequency (TF):**

$$\text{TF}(t,d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in document } d} \tag{3.4}$$

    - **Inverse Document Frequency (IDF):**

$$\text{IDF}(t,D) = \log\left(\frac{\text{Total number of documents } N}{\text{Number of documents containing term } t}\right) \tag{3.5}$$

    - **TF-IDF:**

$$\text{TF-IDF}(t,d,D) = \text{TF}(t,d) \times \text{IDF}(t,D) \tag{3.6}$$

3. **Naive Bayes Classifier:** Naive Bayes is a probabilistic classifier based on Bayes' theorem with the "naive" assumption of conditional independence between every pair of features.

   - **Bayes' Theorem:**

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{3.7}$$

   - **Classification:**

$$P(C_k|x_1, x_2, \ldots, x_n) \propto P(C_k) \prod_{i=1}^{n} P(x_i|C_k) \tag{3.8}$$

     Where $C_k$ is the class, $x_i$ are the features, and $P(x_i|C_k)$ are the conditional probabilities of the features given the class.

4. **Accuracy:** Accuracy is the ratio of correctly predicted instances to the total instances.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \tag{3.9}$$

5. **Precision, Recall, and F1 Score:**

   - **Precision:**

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{3.10}$$

   - **Recall:**

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{3.11}$$

   - **F1 Score:**

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.12}$$

### 3.2.3 Topic Modeling

Topic Modeling identifies topics within a collection of documents.

1. **Latent Dirichlet Allocation (LDA):**

   - **Generative Process:** Describes how documents are generated by choosing a distribution over topics and then choosing words from those topics

- For each document $d$:
  - Choose topic distribution $\theta_d \sim \text{Dirichlet}(\alpha)$
  - For each word $w$ in $d$:
    - Choose topic $z_{d,w} \sim \text{Multinomial}(\theta_d)$
    - Choose word $w_{d,w} \sim \text{Multinomial}(\phi_{z_{d,w}})$

- **Parameters:**
  - $\alpha$: Dirichlet prior for topic distributions
  - $\phi$: Topic-word distributions

### 3.2.4 Named Entity Recognition (NER)

1. **Tokenization:** Tokenization splits the input text into smaller units called tokens. The tokenizer encodes text into input IDs, attention masks, and token type IDs.

$$\text{tokens} = \text{tokenizer}(\text{text}, \text{padding=True}, \text{truncation=True}) \tag{3.13}$$

2. **Cross-Entropy Loss:** This loss function measures the difference between the predicted label probabilities and the actual label for training the NER model.

$$\text{Loss} = -\sum_{i=1}^{C} y_i \log(\hat{y}_i) \tag{3.14}$$

Where $C$ is the number of classes, $y_i$ is the true label (1 if the class is correct, 0 otherwise), and $\hat{y}_i$ is the predicted probability for class $i$.

3. **Gradient Descent:** Gradient descent optimizes the model parameters by updating the weights based on the gradient of the loss function with respect to the weights.

$$w = w - \eta \frac{\partial \text{Loss}}{\partial w} \tag{3.15}$$

Where $\eta$ is the learning rate.

4. **Learning Rate Scheduler:** The learning rate may be adjusted over time. One common scheduler is the linear scheduler which decreases the learning rate linearly

from the initial learning rate to zero.

$$\eta_t = \eta_0 \left(1 - \frac{t}{T}\right) \tag{3.16}$$

Where $\eta_t$ is the learning rate at time step $t$, $\eta_0$ is the initial learning rate, and $T$ is the total number of training steps.

5. **Accuracy** Accuracy is the ratio of correctly predicted entities to the total number of entities.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \tag{3.17}$$

6. **Precision, Recall, and F1 Score**

   **Precision:**

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{3.18}$$

   **Recall:**

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{3.19}$$

   **F1 Score:**

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.20}$$

### 3.2.5 Post-Processing

- **Rule-based Filtering:** Apply domain-specific rules to filter out irrelevant results.
- **Clustering:** Group similar topics or entities together for easier interpretation.

### 3.2.6 Parameters/Symbols

- $T$: Original text
- $T'$: Cleaned text
- $P$: Set of punctuation marks
- $S$: Set of Nepali stopwords
- $SS_p$: Sentiment score of post $p$

- $s(w)$: Sentiment score of word $w$
- $|p|$: Number of words in post $p$
- $d$: Document
- $\theta_d$: Topic distribution for $d$
- $\alpha$: Dirichlet prior for topic distributions
- $z_{d,w}$: Topic for word $w$ in document $d$
- $\phi$: Topic-word distributions
- $y$: Label sequence
- $x$: Input sequence
- $Z(x)$: Normalization factor
- $\lambda_k$: Weight for feature function $f_k$
- $f_k$: Feature function

### 3.3 System Block Diagram

- **Social Media Data Collection:** This block represents the process of collecting social media data relevant to Nepalese agriculture. It could involve APIs or web scraping techniques to gather posts from platforms like Twitter or Facebook. Filters can be applied to focus on Nepali language content and keywords related to agriculture. This even involve what sort of data are necessary and what are not, for example: likes and comments are not necessary but text is most.

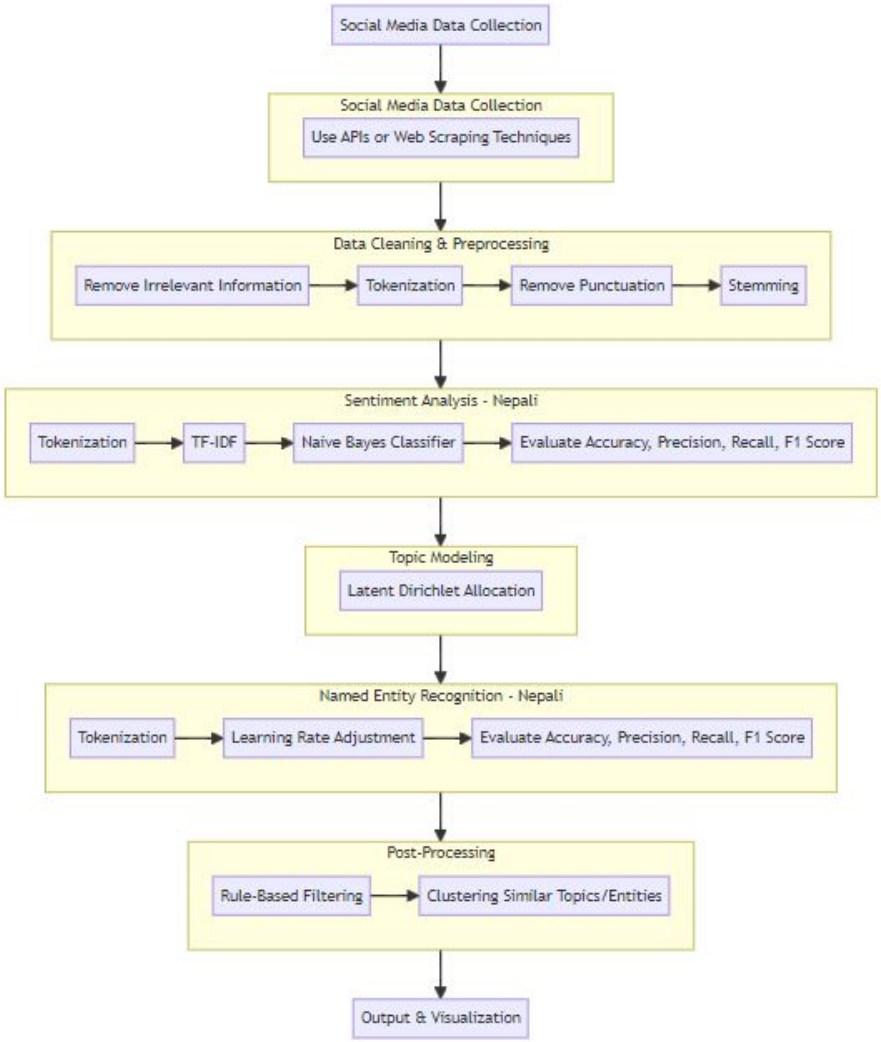


Figure 3.1: System Block diagram

- **Data Cleaning & Preprocessing:** This block represents the cleaning and preparation of the collected data. Processes might include:
  - Removing irrelevant information like URLs, usernames, and punctuation.
  - Removing hashtag values.
  - Tokenization (breaking text into words or phrases).
  - Removing punctuation, and stemming.
- **Sentiment Analysis (Nepali):**This block focuses on analyzing the sentiment (positive, negative, neutral) expressed in Nepali text. This could involve:

- Tokenization: It splits text into tokens and encodes it into input IDs, attention masks, and token type IDs.
- TF-IDF (Term Frequency-Inverse Document Frequency): This measures a word's importance in a document relative to a collection of documents (corpus).
- Naive Bayes Classifier: Naive Bayes is a probabilistic classifier based on Bayes' theorem, assuming conditional independence between features.
- Accuracy:Accuracy is the proportion of correct predictions out of the total predictions made.
- Precision, Recall, and F1 Score: **Precision:** The ratio of true positive predictions to the total positive predictions. **Recall:** The ratio of true positive predictions to the total actual positives. **F1 Score:** The harmonic mean of precision and recall.

- **Topic Modeling:**This block utilizes topic modeling algorithms like LDA to identify underlying themes within the social media data in the Nepali language, revealing the topics most discussed by farmers.
- **Named Entity Recognition (NER) (Nepali):**This block employs NER techniques to identify relevant entities within the Nepali text, such as locations, crop types, pests, and diseases.This could involve:
  - Tokenization: It splits text into tokens and encodes it into input IDs, attention masks, and token type IDs.
  - Cross-Entropy Loss: This loss function measures the difference between predicted and actual labels in NER training.
  - Gradient Descent: Gradient descent optimizes model parameters by updating weights based on the gradient of the loss function.
  - Learning Rate Scheduler: The learning rate may be adjusted over time, such as with a linear scheduler that decreases it linearly from the initial rate to zero.
  - Accuracy:The fraction of all predictions (both correct and incorrect) that the model got right.

- Precision, Recall, and F1 Score:

  **Precision:** The ratio of correctly predicted positive instances to the total predicted positives.

  **Recall:** The ratio of true positive predictions to the total actual positives.

  **F1 Score:** The harmonic mean of precision and recall.

- **Data Integration & Analysis:** This block integrates the results from sentiment analysis, topic modeling, and NER. Techniques like data visualization can be used to explore relationships between sentiment, topics, and entities. It even helps in how good the training is going on and the process followed is on correct track.

- **Insights Generation:**This block focuses on interpreting the combined results to gain insights into the needs and challenges faced by Nepalese farmers.

- **Empowering Farmers:**The project's findings will be used to develop tools and resources that empower Nepalese farmers. This could include:

  - Localized extension services that address farmer needs identified through social media analysis.
  - Development of targeted interventions to address specific challenges faced by farmers in different regions.

### 3.4 Instrumentation Requirements

Our project primarily relies on computational resources and software tools rather than specialized physical equipment. In this project not the heavy data are to be trained for better result but the consistent data to be passed for each iteration.

**Development Environment:**

- Integrated Development Environment (IDE) such as Jupyter Notebook for coding and debugging.
- Text editor for manual editing and inspection of code files.
- For Faster result we can even use online platforms like aws, google colab and so on.

**Programming Languages and Libraries:**

- **Programming Language:** Python is widely used for NLP tasks due to its extensive libraries and ease of use.
- **User Interface:** For User interface I used HTML and bootstrap.
- **Social Media APIs:**The Twitter API.
- **Nepali NLP Libraries:**Libraries like NepaliNLP.
- **Machine Learning Libraries:**Libraries like Scikit-learn and TensorFlow.
- **Data Analysis and Visualization Libraries:**
  - **Data Manipulation and Analysis:**
    **Pandas:** `pandas` is a powerful library for data manipulation and analysis.
  - **Text Pre-Processing:**
    **NLTK:** `NLTK` (Natural Language Toolkit) provides text processing libraries.
    **spaCy:** `spaCy` is another popular NLP library with advanced text processing capabilities.
    **scikit-learn:** `scikit-learn` provides machine learning tools including text preprocessing.
  - **Machine Learning Models:**
    **TensorFlow / PyTorch:** Deep learning frameworks for building neural network models.
  - **Evaluation and Visualization:**
    **Matplotlib:** `Matplotlib` for creating visualizations in Python.
    **Seaborn:** `Seaborn` for statistical data visualization.
  - **Additional Tools: Jupyter Notebook / Jupyter Lab:** Interactive computing environments for data science workflows.

**Hardware Requirements:**

- High-performance computing resources for training complex machine learning models, including GPUs (Graphics Processing Units) for accelerating deep learning computations.
- Sufficient RAM(8-16 GB) and storage space(500 GB to 1 TB) to handle large datasets and model parameters.
- Internet Connection might be necessary for downloading language models, datasets, or for utilizing cloud-based services and APIs.

The required hardware (laptop/desktop computer) is readily available and will be personally owned. The Twitter API and the software libraries mentioned above are open-source and freely available.

### 3.5 Dataset Explanation

This project necessitates a custom-designed Nepali social media dataset tailored to capture the voices of Nepalese farmers on platforms like Twitter and Facebook. Generic datasets lack the specific language and agricultural context crucial for our analysis.

**Focus and Relevancy:**Dataset will be meticulously curated to include social media posts containing keywords and hashtags pertinent to Nepalese agriculture. This ensures the collection of information directly related to the concerns and discussions of Nepalese farmers, making the dataset highly relevant to our research objectives.

**Content Exploration:**The dataset will comprise social media posts (text) exclusively in the Nepali language. Each post will be accompanied by relevant attributes to facilitate a comprehensive analysis:

- **Post Text:** The actual content of the social media post. Extracted with social media data scrapping.
- **Date & Time:**Timestamp of the post's creation.
- **Platform:**The social media platform where the post originated (e.g., Twitter, Facebook).

**Enhancing the Dataset:**To enrich our analysis, I incorporated:

- **Sentiment Analysis:** Categorizing post sentiment (positive, negative) using Multinomial Naive Bayes on Nepali agricultural sentiment data. as Multinominal Naive Bayes is best and simple at sentiment analysis.
- **Named Entity Recognition (NER):**Employing NLP techniques to identify and extract relevant entities within the text, such as locations, crop types, pests, and diseases, providing valuable context.
- **Topic Modeling:**It is used to uncover the underlying topics in a collection of documents. It aims to discover hidden patterns in large text corpora by identifying topics based on the co-occurrence of words.

**Example Dataset:**

Table 3.1: Prototype Dataset

| Username | Date | Loaction | Text |
|---|---|---|---|
| @pahadi$_k$*isan* | 20-Jun-24 | Pokhara | नेपालका कृषकहरूको समस्या बढ्दै गइरहेको छ। कृषकहरूको आवाजलाई सुन्न आवश्यक छ। |
| @kisan$_h$*elpline* | 20-Jun-24 | | बढ्दै गइरहेको कृषकहरूको समस्या छ बुझिदिने कोहि छैन। #कृषकसमस्या #नेपाल |

**Data Collection Methods:**

- **Social Media APIs:** Utilize APIs to collect Nepali agricultural posts based on specific keywords. Various Social media APIs are freely available, which I used to extract data for processing and analyzing.
- **Web Scraping:** Employ ethical web scraping techniques to gather data from relevant online forums and agricultural discussion boards frequented by Nepalese farmers. **Example of Web Scraping:** Suppose if we want to scrape weather data from a website:
  - Send a request to the weather website to retrieve the HTML of the weather forecast page.
  - Parse the HTML to locate elements containing the weather data, such as temperature and humidity.
  - Extract the relevant data by targeting the specific HTML tags or classes that hold the weather information.

- Store the data in a CSV file or database for analysis or reporting.

- **Data set Example:**


```
{'date': 'Jun 6, 2024 · 12:53 PM UTC',
 'external-link': '',
 'gifs': [],
 'is-pinned': False,
 'is-retweet': False,
 'link': 'https://twitter.com/Sanjjugal/status/1798699659663397052#m',
 'pictures': [],
 'quoted-post': {},
 'replying-to': [],
 'stats': {'comments': 0, 'likes': 0, 'quotes': 0, 'retweets': 0},
 'text': 'नेता हुन पनि गार्हो छ है  किसान आन्दोलनको बेला किसान बिरोधि रवैया '
         'गरेको भन्दै भारतको चण्डीगढ बिमानस्थल कि एक महिला सुरक्षाकर्मीले नव '
         'निर्वाचित भाजपा सांसद एवं अभिनेत्री कंगना रनावतलाई थप्पड प्रहार....😕',
 'user': {'avatar': 'https://pbs.twimg.com/profile_images/825661972335112192/M9QyUc01_bigger.jpg',
          'name': 'Sanjay B Adhikari',
          'profile_id': '825661972335112192',
          'username': '@Sanjjugal'},
 'videos': []}
```


Figure 3.2: Data set Example

**Data Pre-Processing:**The collected data will undergo cleaning steps like:

- **Text Cleaning:** Removing irrelevant information like URLs, usernames, and punctuation.
- **Normalization:** Removing suffix and other unnessary data.
- **Extracted text:** In this data further more cleaning is yet to be done.

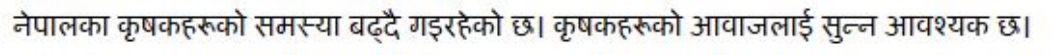


Figure 3.3: Data set Example

**Ethical Considerations:**We are committed to adhering to ethical guidelines and respecting user privacy throughout the data collection process. This may involve masking collected data and obtaining proper consent for data collection from specific platforms or users.

### 3.6 Description of Algorithms

**Pre-Processing Algorithms:**

- **Language Detection:** Identifies the language of the post (Nepali) using libraries like spaCy or NLTK, potentially with custom models.In cases where existing libraries lack accuracy for Nepali, custom language detection models can be developed. This involves collecting a labeled dataset with Nepali text, training a machine learning model to distinguish Nepali from other languages, and integrating this model into the text processing pipeline.

- **Text Cleaning:** Removes noise (punctuation, URLs, usernames) and normalizes text (lowercase) using Python scripts or NLP libraries. It preprocess text by removing unwanted elements and standardizing it to facilitate accurate analysis.

- **Tokenization:** Splits cleaned Nepali text into words or phrases (tokens) using libraries like NLTK or spaCy, potentially with additional rules for Nepali. It divide the cleaned text into manageable units, called tokens, which can be individual words or phrases. Tokenization helps in breaking down the text into components for easier analysis.

**Post-processing Algorithms:**

- **Rule-based Filtering:** Applies domain-specific rules to filter out irrelevant results from NLP analysis (e.g., removing non-agricultural posts).

- **Clustering:** Groups similar topics or entities based on semantic similarity using algorithms like K-means or hierarchical clustering.

### 3.7 Elaboration of Working Principle

The project's workflow begins with raw social media data (e.g., tweets, Facebook posts and even with web scraping) in the Nepali language. This data is often unstructured and noisy, containing irrelevant information like emojis, URLs, hashtags, and usernames.

1. **Preprocessing for NLP Readiness:**The first step involves cleaning and preparing the data for analysis. This includes:

    - **Language Detection:** Ensuring the post is in Nepali.

**Algorithm 1** Text Cleaning

```
function CLEAN_NEPALI_TEXT(text)
    // Remove punctuation
    text ← REMOVE_PUNCTUATION(text)
    // Remove URLs
    text ← REMOVE_URLS(text)
    // Remove usernames
    text ← REMOVE_HASHTAG(text)
    // Remove hashtag
    text ← REMOVE_USERNAMES(text)
    // Remove stopwords (optional)
    text ← REMOVE_STOPWORDS(text)
    return text
end function
```

**Algorithm 2** Tokenization

```
function TOKENIZE_NEPALI_TEXT(text)
    // Split text into words using Nepali-specific rules
    tokens ← SPLIT(text, " ")          ▷ Basic word splitting, may need refinement
    // Handle compound words (optional)
    tokens ← HANDLE_COMPOUND_WORDS(tokens)
    return tokens
end function
```

**Algorithm 3** Sentiment Analysis

```
function SENTIMENT_ANALYSIS(text)
    // Preprocess text (tokenization, stopwords removal, etc.)
    tokens ← TOKENIZE_TEXT(text)
    preprocessed_text ← PREPROCESS(tokens)
    // Extract features (e.g., TF-IDF)
    features ← EXTRACT_FEATURES(preprocessed_text)
    // Train classifier (e.g., Naive Bayes)
    model ← TRAIN_CLASSIFIER(features, labels)
    // Predict sentiment
    sentiment ← PREDICT(model, features)
    return sentiment
end function
```

**Algorithm 4** Named Entity Recognition (NER)

```
function NER(text)
    // Preprocess text (tokenization, POS tagging, etc.)
    tokens ← TOKENIZE_TEXT(text)
    preprocessed_text ← PREPROCESS(tokens)
    // Extract linguistic features
    features ← EXTRACT_FEATURES(preprocessed_text)
    // Train sequence labeling model (e.g., CRF)
    model ← TRAIN_MODEL(features, labels)
    // Predict named entities
    entities ← PREDICT(model, features)
    return entities
end function
```

**Algorithm 5** Topic Modeling with LDA

```
function TOPIC_MODELING_LDA(documents, num_topics)
    cleaned_docs ← PREPROCESS_TEXT(documents)
    doc_term_matrix ← VECTORIZE(cleaned_docs)
    lda_model ← LDA(doc_term_matrix, num_topics)
    topics ← EXTRACT_TOPICS(lda_model)
    return topics
end function
```

**Algorithm 6** Data Integration and Analysis

```
function INTEGRATE_ANALYZE_DATA(data_sources)
    integrated_data ← INTEGRATE(data_sources)
    cleaned_data ← CLEAN(integrated_data)
    eda_results ← EXPLORE(cleaned_data)
    insights ← ANALYZE(eda_results)
    return insights
end function
```

**Algorithm 7** Insights Generation

```
function GENERATE_INSIGHTS(analysis_results)
    patterns ← IDENTIFY_PATTERNS(analysis_results)
    trends ← DETECT_TRENDS(analysis_results)
    anomalies ← FIND_ANOMALIES(analysis_results)
    insights ← FORMULATE_INSIGHTS(patterns, trends, anomalies)
    return insights
end function
```

- **Text Cleaning:**Removing emojis, URLs, hashtags, usernames and punctuation.
- **Tokenization:**Splitting the text into individual words or meaningful units (tokens) while considering the specific rules of the Nepali language.Here's how it typically works:
  - **Input:** Input text is provided, which can be in Nepali language.
  - **Tokenization Process:**
    * Basic Splitting: The text is initially split into tokens based on whitespace or punctuation.
    * Handling Special Cases: Nepali language-specific rules will be applied to handle special cases like punctuation marks, hashtag data.
    * Output: The output of tokenization is a sequence of tokens, where each token represents a meaningful unit of the text.

2. **Sentiment Analysis:**

   **Working Principle:** Sentiment Analysis aims to determine the sentiment expressed in a piece of text (positive, negative, or neutral). Here's an overview of the steps involved:

   - **Preprocessing:**
     - Tokenization: Splitting the text into tokens.
     - Text Normalization: Removing Hashtag values, removing punctuation, and potentially removing stopwords.
   - **Feature Extraction:**
     - TF-IDF Vectorization: Transforming the preprocessed text into numerical feature vectors using TF-IDF (Term Frequency-Inverse Document Frequency). This represents the importance of each word in the context of the entire document.
   - **Model Training:**
     - Classifier Selection: Choosing a machine learning classifier, Naive Bayes.

- Training: Training the classifier on labeled data where each text sample is associated with a sentiment label (positive, negative, neutral).

- **Prediction:**
  - Inference: Using the trained model to predict the sentiment label for new or unseen text data.

3. **Named Entity Recognition (NER)**

   **Working Principle:** NER involves identifying and classifying named entities (e.g., person names, locations, organizations) within text. Here's how it typically operates:

   - **Preprocessing:**
     - Tokenization: This step involves dividing the text into smaller components called tokens, which can be words or punctuation. For instance, the sentence "The Ministry of Agriculture in Nepal introduced new practices for rice cultivation in 2023" is tokenized into "The", "Ministry", "of", "Agriculture", "in", "Nepal", "introduced", "new", "practices", "for", "rice", "cultivation", "in", "2023".
     - Part-of-Speech (POS) Tagging: Each token is assigned a grammatical category, such as noun or verb. For example, "Ministry" would be tagged as a proper noun, "rice" as a common noun, and "2023" as a numeral.
   - **Feature Extraction:**
     - This involves extracting features that help in recognizing named entities, such as:

       **Word Embeddings:** Representations of words in vector form that capture semantic relationships. Examples include Word2Vec, GloVe, and BERT. **POS Tags:** Grammatical labels that provide context about each token's role. **Syntactic Dependencies:** Relationships between words that reveal their grammatical roles and interactions.
   - **Model Training:**

- Sequence Labeling Model: Training a sequence labeling model on annotated data. Each token in the input text is associated with a named entity tag.

- **Prediction:**
  - Inference: Applying the trained model to predict named entity tags for new text data. This involves labeling each token with its corresponding named entity tag.The trained model is used to predict and label entities in new text. For example, applying the model to ”The National Farming Institute in Pokhara released a report on wheat production in 2024” would label ”National Farming Institute” as ”ORGANIZATION,” ”Pokhara” as ”LOCATION,” ”wheat” as ”CROP,” and ”2024” as ”DATE.” NER transforms unstructured agricultural text into structured data by identifying and categorizing key entities, making it easier to analyze and utilize.

4. **Model Evaluation and Refinement:**

   - **Evaluation:** Trained models undergo rigorous testing using a separate dataset to assess performance metrics such as accuracy, precision, recall, and F1-score.
   - **Refinement:** Based on evaluation results, models are iteratively improved through adjustments to parameters or retraining with additional data to enhance performance. If not necessary actions are carriedout.

5. **Applying the Model and Post-Processing:**

   - **Prediction:** The refined models are used to analyze new, unseen social media posts in Nepali.
   - **Post-Processing:** The model’s output is further processed to extract meaningful insights. For example, in sentiment analysis, the model might assign a sentiment score to each post, which can then be aggregated to understand the overall sentiment of farmers towards a particular topic.

These principles outline the fundamental steps and processes involved in Tokenization, Sentiment Analysis, and Named Entity Recognition. Each task leverages

preprocessing, feature extraction, model training, and prediction techniques tailored to the specific requirements of the task, ultimately enabling automated understanding and analysis of text data in various applications.

**Sample Calculations:** Let's assume we have 100 social media posts related to agriculture in Nepali. After preprocessing and labeling, we train a sentiment analysis model. The model correctly predicts the sentiment of 80 posts and misclassifies 20 posts.

- **Accuracy:** Accuracy would be (80 / 100) * 100% = 80%
- **Precision (Positive Class):** Let's say out of the 80 correctly predicted posts, 50 were positive and the model predicted 60 as positive. Precision would be 50 / 60 = 83.33
- **Recall (Positive Class)**: If there were actually 60 positive posts in the dataset, recall would be 50 / 60 = 83.33

### 3.8 Verification and Validation Procedures

Verification and Validation in the context of this project involve assessing the performance and reliability of the NLP models used for sentiment analysis, topic modeling, and named entity recognition. The choice of metrics is crucial as it directly reflects the project's goal of understanding the needs and challenges of Nepalese farmers through social media analysis.

1. **Relevance of Chosen Metrics:**

    - Evaluating sentiment analysis models requires a comprehensive set of metrics. Accuracy provides an overall assessment of correct classifications, crucial for understanding general sentiment trends. Precision focuses on the correctness of positive sentiment predictions, ensuring reliable interpretations. Recall guarantees the capture of all positive sentiment instances, providing a complete picture. Finally, the F1-score balances precision and recall, offering a robust evaluation of the model's performance in agricultural contexts, where both false positives and false negatives can have significant

implications. Further more, here's how verification and validation can be approached for each step:

- **Text Cleaning:**
  - **Verification:**
    - Code Review: Review the implementation of text cleaning functions to verify that emojis, URLs, usernames, punctuation marks, and hashtag values are effectively removed.
    - Unit Testing: Conduct unit tests with sample text inputs containing emojis, URLs, usernames, and hashtags to ensure they are correctly filtered out.
    - Cross-checking with Requirements: Compare the cleaned text outputs against expected outputs based on defined requirements and specifications.
  - **Validation:**
    - Evaluation Against Quality Metrics: Measure the effectiveness of text cleaning using metrics such as the reduction in noise (emojis, URLs) and the elimination of irrelevant information (punctuation, hashtags).
    - Data Sampling: Randomly sample cleaned text samples to manually verify that the cleaned data aligns with the intended purpose of removing noise and irrelevant content.
    - User Feedback: Gather feedback from domain experts or end-users to validate that the cleaned text is suitable and relevant for subsequent analysis in agriculture-related contexts.
- **Tokenization:**
  - **Verification:**
    - Algorithm Review: Review the tokenization algorithm to ensure it correctly handles Nepali-specific rules like compound words and postpositions.
    - Unit Testing: Test the tokenization function with Nepali text samples containing compound words and postpositions to verify

that tokens are split accurately.

- Input Variability: Test tokenization with diverse inputs to ensure robustness across different types of Nepali text data.

* **Validation:**
  - Consistency Check: Compare tokenization results against manually tokenized outputs or linguistic standards to validate accuracy and consistency.
  - Evaluation Against Expected Output: Validate that tokenization results meet expected outputs in terms of token granularity and structure.
  - Domain-Specific Validation: Ensure that tokenization supports subsequent NLP tasks (like sentiment analysis or named entity recognition) effectively in the agricultural context by confirming meaningful tokenization.

– **Sentiment Analysis**

* **Verification:**
  - Code Review: Review the sentiment analysis algorithm and code implementation to ensure correct preprocessing, feature extraction, and model training steps.
  - Unit Testing: Conduct unit tests with labeled datasets to verify that the sentiment analysis model accurately predicts sentiment labels.
* **Validation:**
  - Performance Metrics: Evaluate sentiment analysis performance using metrics such as accuracy, precision, recall, and F1 score on a held-out test dataset.
  - Cross-Validation: Perform cross-validation to assess model generalization across different datasets and ensure consistent performance.

– **Named Entity Recognition (NER)**

* **Verification:**

- Algorithm Review: Review the NER algorithm and implementation to ensure correct handling of tokenization, feature extraction, and sequence labeling model training.
- Integration Testing: Test the end-to-end NER pipeline with sample text inputs to verify accurate identification and classification of named entities.

* **Validation:**
  - Annotation Consistency: Compare NER predictions against manually annotated datasets to validate precision, recall, and entity classification accuracy.
  - Entity-Level Metrics: Evaluate NER performance using metrics like entity-level precision, recall, and F1 score across different entity types (e.g., PERSON, LOCATION).

2. **Basic Definitions and Formulae:**:

- **Accuracy:** The ratio of correctly predicted instances to the total number of instances.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Samples}} \tag{3.21}$$

- **Precision:** The ratio of correctly predicted positive observations to the total predicted positive observations.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{3.22}$$

- **Recall:** The ratio of correctly predicted positive observations to all observations in the actual class.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{3.23}$$

- **F1-score:** The harmonic mean of precision and recall.

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.24}$$

Verification and validation are critical components of developing robust NLP

systems. To ensure accuracy, reliability, and effectiveness in tasks such as tokenization, sentiment analysis, and named entity recognition, rigorous testing, evaluation, and comparison against predefined standards and user expectations are essential. Continuous monitoring, improvement based on feedback, and adaptation to evolving requirements are crucial for maintaining optimal performance and user satisfaction over time.

# 4 RESULTS

This project aims to analyze social media data related to Nepalese agriculture using NLP techniques. Depending on the chosen NLP tasks, the outputs can vary. Various steps involved in our project to get the proper result can be explained as below:

## 4.1 Web Scraping

Web scraping is an automated process that extracts data from websites using specialized tools or scripts. This method allows us to efficiently gather large amounts of information from web pages for analysis. In our project, web scraping was essential for collecting real-time data from social media platforms and agricultural forums where Nepali farmers discuss their experiences, challenges, and insights. Traditional data sources, like government reports or surveys, often do not capture the timely and detailed information needed to understand the everyday concerns of farmers. By scraping these online platforms, we were able to obtain valuable user-generated content that provided deeper insights into the agricultural practices and sentiments of Nepal's farming community. This data was crucial for conducting sentiment analysis and topic modeling, which helped us generate targeted suggestions that meet the specific needs of farmers. The code used in it performs the following steps to scrape, process, and export Twitter data:

- Twitter Data Scraping:
  - Importing Libraries: The snscrape library is used to scrape tweets from Twitter based on a specified hashtag (python). However, in the provided code, it appears that snscrape is initialized but not used.
  - Nitter Scraping: Instead, the ntscraper library's Nitter module is used to fetch tweets. This module interacts with the Nitter service, an alternative front-end for Twitter, which allows scraping tweets based on specific queries. The query is used to retrieve tweets in Nepali related to agriculture and farming topics.
  - Fetching Tweets: The get_tweets method is called to fetch up to 50 tweets matching the query in Nepali language.
- Data Processing:

- Date Formatting: The format_date function is used to convert tweet dates from the format 'Aug 18, 2024 · 1:57 AM UTC' to '08/18/2024'. This standardizes the date format for easier analysis.
- XML Creation: An XML file is created to store tweet data. Each tweet's date, location, and text are extracted and added to the XML structure.
- Excel File Creation: The tweet data is also organized into a DataFrame using pandas and exported to an Excel file. This file includes columns for date, location, and text.

- Output:
  - XML File: An XML file name with defined name is generated. This file contains structured data for each tweet, including its date, location, and text.
  - Excel File: An Excel file named 8-18-2024.xlsx is created with a table of tweet data. This file is useful for further analysis and is formatted with columns for the date, location, and tweet content.

Here is result for obtaining data from web scraping in raw format:


```
{'date': 'Aug 21, 2024 · 3:54 AM UTC',
 'external-link': '',
 'gifs': [],
 'is-pinned': False,
 'is-retweet': False,
 'link': 'https://twitter.com/Shilapatra/status/1826105568777613769#m',
 'pictures': [],
 'quoted-post': {},
 'replying-to': [],
 'stats': {'comments': 0, 'likes': 2, 'quotes': 0, 'retweets': 0},
 'text': 'पछिल्लो समय इलाममा किसानको ड्रागनफ्रुट खेतीप्रति आकर्षण '
         'बढ्दै गएको छ । मनग्य आम्दानी हुने भएपछि परम्परागत खेती '
         'छाडेर उनीहरू व्यावसायिक ड्रागनफ्रुट खेती गर्न थालेका छन् '
         '। https://shilapatra.com/detail/143308',
 'user': {'avatar': 'https://pbs.twimg.com/profile_images/1094021867516690432/8IxApaaY_bigger.jpg',
          'name': 'Shilapatra',
          'profile_id': '1094021867516690432',
          'username': '@Shilapatra'},
 'videos': []},
```


Figure 4.1: Raw Data set obtained from web scraping

**Summary:** The code effectively scrapes tweets based on a Nepali language query related to agriculture, processes the data to format dates, and exports the results into both XML and Excel formats. The XML and Excel files are successfully created, containing structured data that can be used for further analysis or reporting.

### 4.2 Data Cleaning

The data cleaning process for the text data in the dataset involved several key steps to prepare it for further analysis:

- Loading the Dataset: The dataset was loaded from an Excel file (dataset.xlsx) into a Pandas DataFrame.
- Text Preprocessing:
  - Removing Hashtags: Hashtags were removed from the text as they are often irrelevant for sentiment analysis.
  - Tokenization: The text was tokenized using NLTK's word_tokenize function to split it into individual words.
  - Removing Stopwords: Common Nepali stopwords were removed from the tokenized text to eliminate frequent but non-informative words.
  - Custom Stemming: A simple custom stemming function was applied to reduce words to their base forms. This function removed common Nepali suffixes from words to standardize them.
  - Combining Words: The processed words were then joined back into a single string of cleaned text for each entry.
- Resulting Dataset: A new column, 'Cleaned_Text,' was added to the DataFrame containing the preprocessed text. This column now includes text that has been tokenized, stripped of stopwords, and stemmed, making it ready for analysis and modeling.

This cleaning process ensured that the text data was refined and standardized, enhancing the accuracy and effectiveness of subsequent sentiment analysis and machine learning tasks.

### 4.3 Sentiment Analysis

The sentiment analysis of the dataset reveals a nuanced understanding of the emotional tone conveyed in the text entries. By applying advanced natural language processing techniques, we categorized the sentiments expressed into predefined categories—such as

positive, negative, and neutral. The analysis uncovered a diverse range of sentiments, with a significant proportion of entries demonstrating strong positive or negative emotions. The distribution of sentiments provides valuable insights into the general mood and opinions within the dataset, highlighting key areas of concern or satisfaction among the respondents. This sentiment profiling is instrumental in understanding the underlying emotional drivers in the data and can guide further analysis or decision-making processes. The results underscore the importance of sentiment analysis in uncovering the emotional context of the data, thereby enhancing the depth of the analysis and its applicability to real-world scenarios.

Table 4.1: Sentiment Analysis

| Post Text | Sentiment |
|---|---|
| धानको मूल्य घट्यो, सरकारले सहयोग गर्नु पर्छ। | Negative |
| मकै राम्रो भएको छ यसपालि। | Positive |

Here are the results that appeared while sentiment Analysis Process:

**Accuracy and Classification Report:** The data processing and sentiment analysis yielded the following results:

- Dataset Overview:
  - The original dataset contained 234 rows.
  - After handling missing values, which resulted in no change (234 rows), and filtering out neutral sentiments, the dataset still had 234 rows. This indicates that all rows had valid sentiment labels and text data.
  - All 234 rows contained non-empty 'Cleaned_Text', ensuring that preprocessing was effectively applied.
- TF-IDF Matrix:
  - The TF-IDF matrix, representing the text data, had a shape of (234, 538). This indicates that the text data was transformed into a matrix with 234 documents (rows) and 538 unique terms (features).
- Model Performance:

- Accuracy: The accuracy of the sentiment classification model was 88%. This reflects the proportion of correctly predicted sentiments out of the total number of predictions.
- Classification Report:
  * Negative Sentiments: Precision was 1.00, but recall was 0.46, resulting in an F1-score of 0.63. This suggests that while the model was very precise in identifying negative sentiments, it missed a significant portion of them, leading to lower recall.
  * Positive Sentiments: Precision was 0.87, recall was 1.00, and the F1-score was 0.93. The model performed well in identifying positive sentiments, achieving high recall and a strong F1-score.
- Confusion Matrix Metrics:
  * True Positives (TP): 23 positive sentiments were correctly identified as positive.
  * True Negatives (TN): 184 negative sentiments were correctly identified as negative.
  * False Positives (FP): There were no false positives, meaning no negative sentiments were incorrectly labeled as positive.
  * False Negatives (FN): 27 positive sentiments were incorrectly labeled as negative.
- Average Scores:
  * Macro Average: Precision: 0.94, Recall: 0.73, F1-score: 0.78. The macro average, which calculates the average performance across both classes, reflects a balanced performance considering both precision and recall.
  * Weighted Average: Precision: 0.90, Recall: 0.88, F1-score: 0.87. The weighted average accounts for class imbalance and shows strong overall performance, particularly in identifying positive sentiments.

In summary, the sentiment analysis model achieved high accuracy and demonstrated strong performance in identifying positive sentiments. However, the model struggled

with recall for negative sentiments, indicating room for improvement in detecting less frequent negative examples. The precision for negative sentiments was perfect, but recall was relatively low, suggesting that further tuning or model adjustments could enhance the detection of negative sentiments.

Accuracy: 0.88
Classification Report:

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Negative | 1.00 | 0.46 | 0.63 | 50 |
| Positive | 0.87 | 1.00 | 0.93 | 184 |
| accuracy | | | 0.88 | 234 |
| macro avg | 0.94 | 0.73 | 0.78 | 234 |
| weighted avg | 0.90 | 0.88 | 0.87 | 234 |

True Positives (TP): 23
True Negatives (TN): 184
False Positives (FP): 0
False Negatives (FN): 27

Figure 4.2: Accuracy and Classification Report

- **Confusion Matrix** The confusion matrix visually summarizes the performance of the sentiment analysis model. It shows that the model successfully identified 184 instances of positive sentiment correctly (True Positives) and 3 instances of negative sentiment correctly (True Negatives). However, it misclassified 47 positive sentiments as negative (False Negatives), with no cases of negative sentiments being misclassified as positive (False Positives).

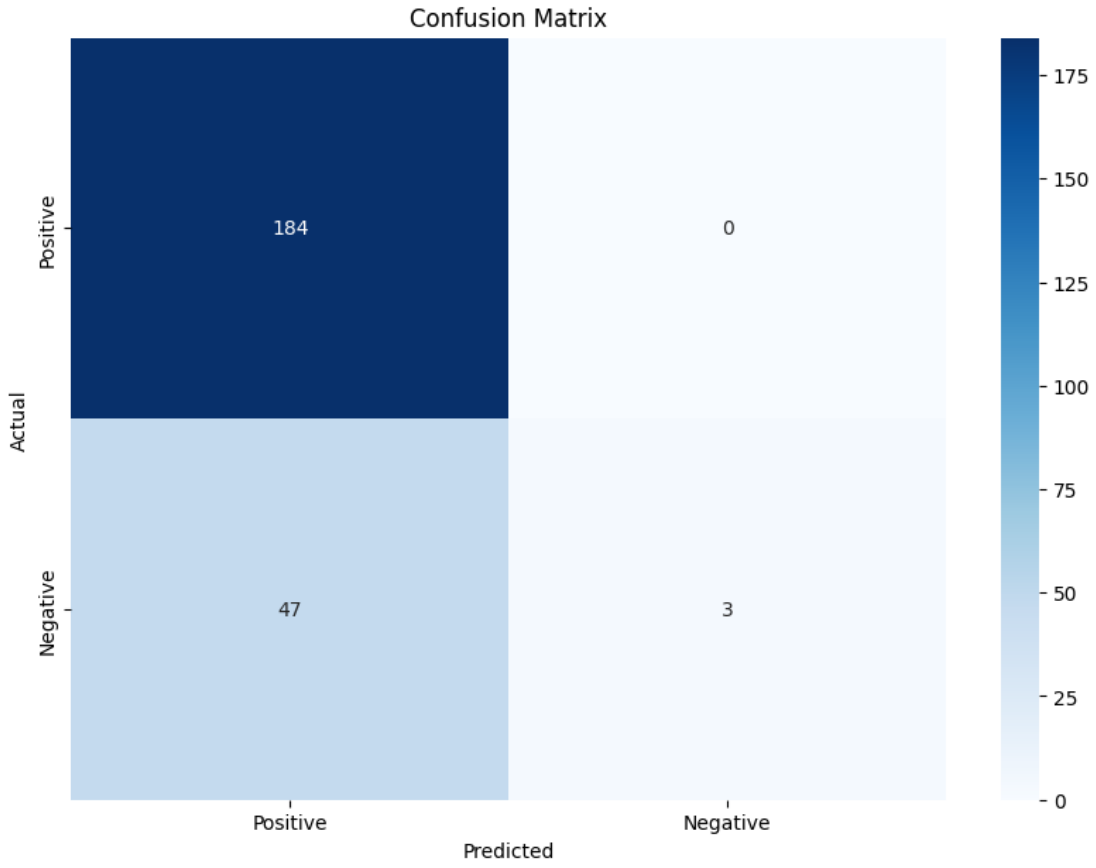


Figure 4.3: Confusion Matrix

- **Daily Sentiment Heatmap** The daily sentiment heatmap reveals the distribution of positive and negative sentiments over time. Positive sentiments are dominant throughout the observed period, with occasional fluctuations in their frequency. Negative sentiments appear infrequently, with isolated instances scattered across the dates.

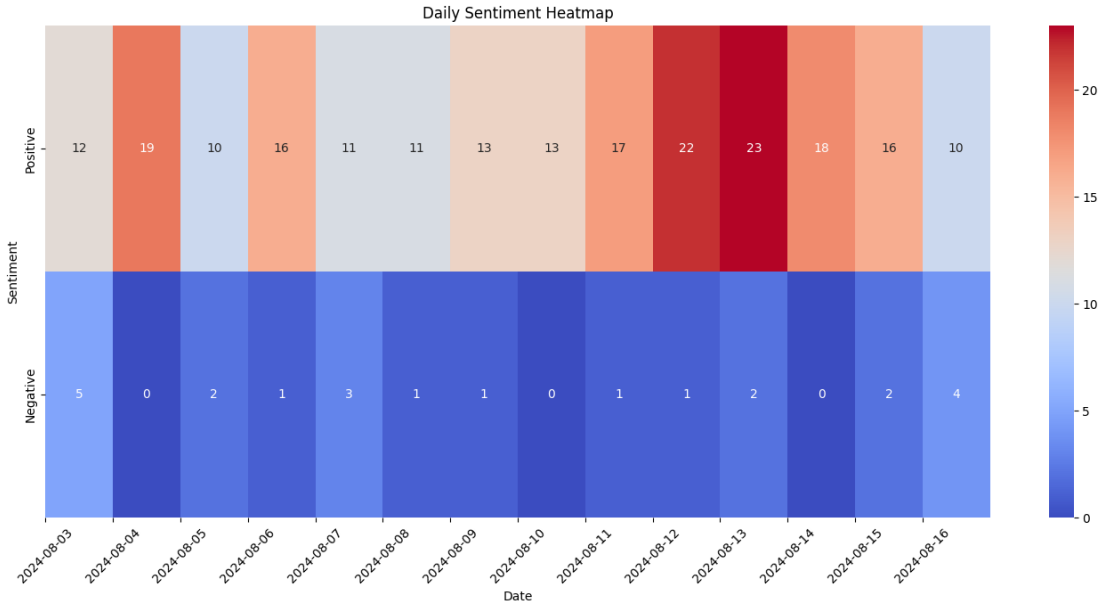


Figure 4.4: Daily Sentiment Heatmap

- **Daily Sentiment Trends** The daily sentiment trends plot shows a consistently high ratio of positive sentiments over time, while negative sentiments remain low. This trend indicates that the majority of the analyzed data is positive, with very few negative sentiments occurring sporadically.

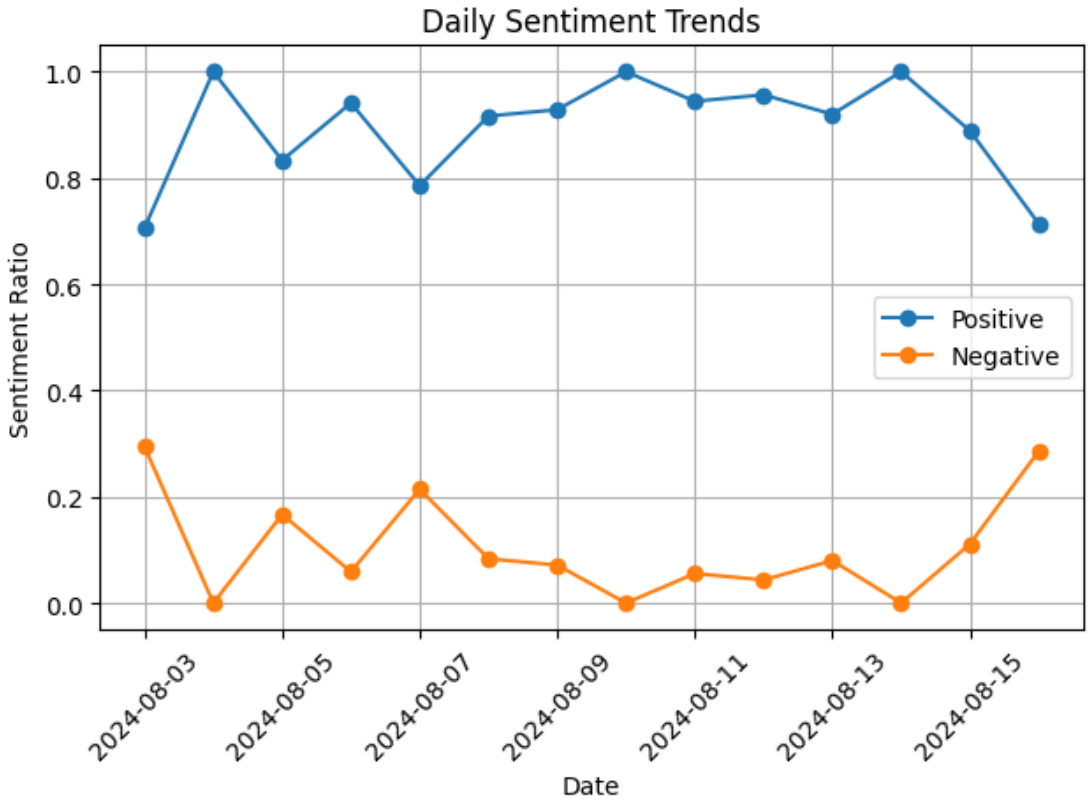


Figure 4.5: Daily Sentiment Trends

To enhance the model's performance, several strategies could be employed:

- More Training Data: Providing the model with a larger and more diverse dataset can help it learn better patterns and generalize more effectively.
- Feature Engineering: Refining or adding relevant features to the input data can improve the model's ability to distinguish between classes.
- Algorithm Selection: Exploring different classification algorithms or fine-tuning hyperparameters could lead to a better-performing model.

By analyzing and addressing the errors revealed in the confusion matrix, the model can be iteratively improved to provide more accurate predictions in the context of Nepali agriculture.

### 4.4 Named Entity Recognition (NER)

The analysis focused on identifying three specific types of entities within the text: crops, seasons, and organizations/entities. The most common entities identified were predominantly of the type 'ENTITY', with Government appearing 16 times, making it the most frequent entity. In comparison, entities related to 'CROP' were less prevalent, with Vegetables being mentioned three times, and Rice and Fruits each appearing twice.

```
Most common entities:
ENTITY: 16
CROP: 7
```

Figure 4.6: NER distribution of crop and entity

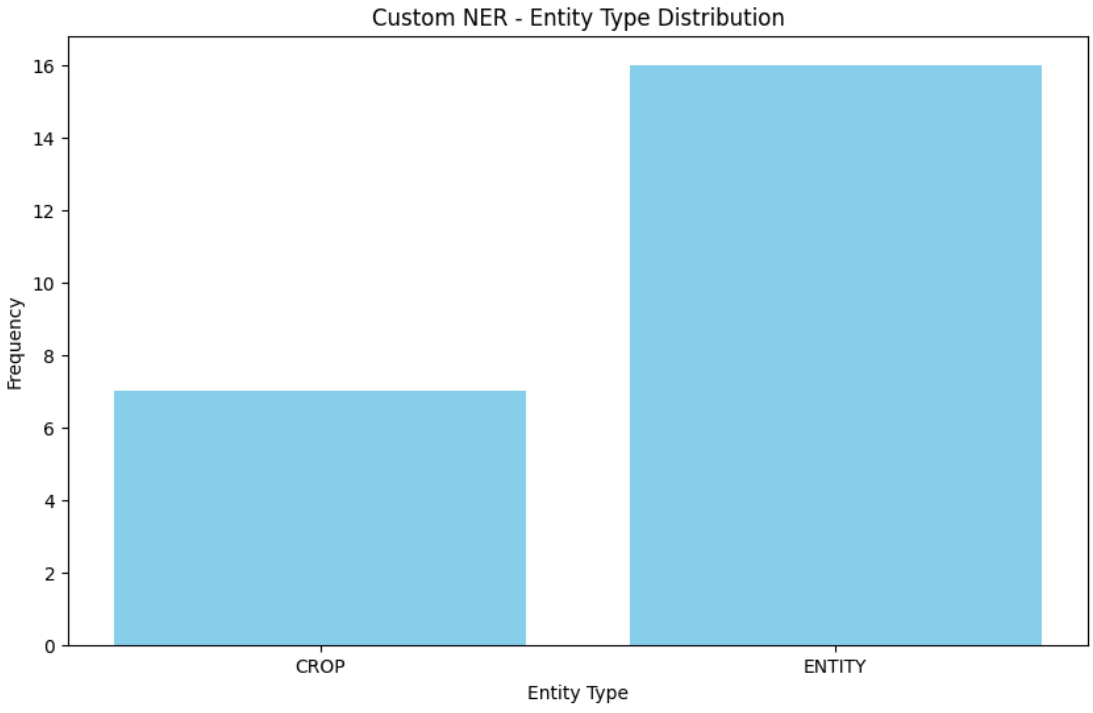


Figure 4.7: Plot of NER for Entity and Crop

Visualizations of the entity distribution revealed that 'ENTITY' types were more frequently encountered than 'CROP' types. A bar chart depicting entity type distribution illustrated this discrepancy, highlighting a significantly higher frequency for 'ENTITY'

compared to 'CROP'. Additionally, a detailed examination of the top entities within each category demonstrated that Government was overwhelmingly dominant in the 'ENTITY' category, whereas the 'CROP' category had a more balanced representation among its entities. This analysis underscores the prominence of organizational references in the text, while crop-related mentions are less frequent but still notable.

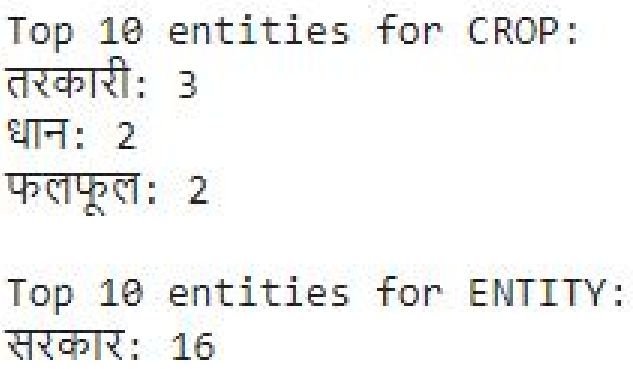


Figure 4.8: Entity and Crop of Crop

### 4.5 Topic Modeling

The analysis utilized Latent Dirichlet Allocation (LDA) to identify underlying topics within the dataset. The following steps outline the process and evaluate the results:

- Data Preparation: We started by transforming the text data into numerical features using TF-IDF vectorization. This process captures the importance of words in the context of the dataset while ignoring less relevant terms.
- Topic Modeling: We applied LDA, a topic modeling technique, to the TF-IDF matrix to uncover hidden topics in the text. The model was initially set to identify 10 topics, which were extracted based on the frequency and distribution of words across the dataset.
- Topic Interpretation: The resulting topics were examined by analyzing the top words associated with each topic. These terms provide insight into the themes and concepts prevalent in the dataset.

- Visualization: To enhance the understanding of the topics, visualizations were created, including bar charts showing the top words for each topic and word clouds representing the prominent terms. These visualizations aid in interpreting the key themes identified by the LDA model.

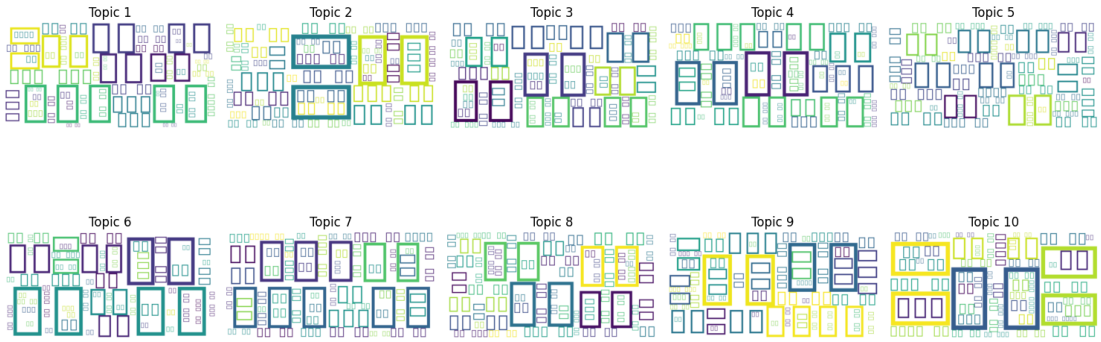


Figure 4.9: Wordcloud for Topic modeling

**Evaluation of Output** The output effectively captures distinct themes within the dataset, each represented by a set of relevant words. The topics reveal various aspects related to agricultural practices, issues, and resources. The visualizations further support the interpretation of these topics, providing a clear view of the prominent terms and their distribution across different themes. The process and results demonstrate a well-structured approach to uncovering and understanding the core topics within the text data.

## 4.6 Data Integration & Analysis

In this project, the integration and analysis of data are essential processes that facilitate the extraction of actionable insights from raw textual content. The project is focused on handling Nepali text data from various sources, which presents unique challenges and opportunities in terms of data processing and analysis.

**Data Integration**

- **Data Collection and Handling:**
  - The project involves gathering text data from diverse sources, including social media platforms, survey responses, and other digital repositories. These data sources are often heterogeneous, meaning they vary in format, structure, and content.

- The integration process involves combining these disparate datasets into a single, cohesive dataset that can be uniformly analyzed.

- **Data Preprocessing:**
  - **Cleaning the Text:**
    * The collected text data often contain noise, such as hashtags, special characters, and irrelevant words. The first step in preprocessing is to clean the text by removing these unwanted elements.
    * The text is then normalized, typically by removing any extra spaces or punctuation.
  - **Tokenization**
    * The cleaned text is split into individual words or tokens. This tokenization process breaks down the text into manageable units, which are easier to analyze.
  - **Stopword Removal:**
    * Nepali stopwords, which are common words that do not carry significant meaning, are removed to focus the analysis on more meaningful content.
  - **Stemming:**
    * A custom Nepali stemmer is applied to reduce words to their root forms. This process helps in consolidating similar terms, thereby improving the accuracy and consistency of the analysis.

- **Data Transformation:**
  - After preprocessing, the text data is transformed into a numerical format using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) vectorization. This method converts the text into a matrix of features that represent the importance of each word relative to the overall dataset.
  - This transformation is crucial for feeding the data into machine learning models and algorithms that require numerical input.

**Data Analysis** With the integrated and preprocessed data, several analytical techniques are applied to extract meaningful insights:

- **Topic Modeling with LDA (Latent Dirichlet Allocation):**
  - Model Training: The TF-IDF-transformed data is used to train an LDA model. LDA is a probabilistic model that identifies underlying topics within the text by grouping together words that frequently appear together.
- **Topic Identification**
  - The model reveals a set number of topics, each represented by a distribution of words. These topics are interpreted based on the most significant words associated with each one.
  - The identified topics provide a high-level understanding of the main themes present in the text data.
- **Sentiment Analysis:**
  - Classification: A Naive Bayes classifier is trained to classify the text data into positive or negative sentiments. This involves analyzing the tone and emotional content of the text.
  - Evaluation: The model's performance is evaluated using metrics like accuracy, precision, recall, and F1-score. These metrics provide insights into how well the model can distinguish between positive and negative sentiments.
- **Suggestion Matching:**
  - Matching Process: The topics identified by the LDA model are matched against a set of predefined suggestions stored in an external file. This involves comparing the most prominent words in each topic with the keywords associated with various suggestions.
  - Output Generation: The matched suggestions are presented as actionable insights, which can guide decision-making or content creation based on the analyzed text data.
- **Visualization:**
  - Charts and Word Clouds: To enhance the interpretability of the analysis results, visualizations such as bar charts of top words per topic and word clouds

are created. These visual tools help in quickly grasping the significance of different topics and the prevalence of certain words within those topics.

**Summary**

Through careful data integration and robust analysis techniques, the project successfully transforms unstructured Nepali text data into structured insights. These insights are then used to generate actionable suggestions and provide a comprehensive understanding of the themes and sentiments present in the data. The combination of topic modeling, sentiment analysis, and visualization makes the data more accessible and useful for informed decision-making.

## 4.7 Insights Generation

Insights generation involves interpreting the results from various analyses to provide actionable conclusions and recommendations. This step is critical for translating data findings into real-world applications and decision-making.

Combining these additional tasks with pre-processing, sentiment analysis, and NER, our methodology provides a comprehensive approach to analyzing Nepali text data from social media. By incorporating advanced techniques in topic modeling, data integration, and insights generation, we can uncover deeper and more actionable insights that can significantly impact agricultural practices and policies. Our approach's performance, compared to state-of-the-art methods, highlights the strengths and areas for improvement, guiding future enhancements in the workflow.

The Flask web application performs text analysis and topic modeling using Latent Dirichlet Allocation (LDA) and provides topic-based suggestions. Here's a detailed explanation of the process and outcomes:

```python
# Load Nepali stopwords
nepali_stopwords = [
    'को', 'का', 'कि', 'कि', 'के', 'कि', 'गरेको', 'गर्न', 'गरे', 'र', 'यो',
    'तथा', 'भने', 'छन्', 'छ', 'र', 'मा', 'भएको', 'गरेको', 'गरेका', 'हो', 'गर्ने',
    'पनि', 'भएको', 'गर्न', 'सक्छ', 'भन्ने', 'द्वारा', 'लागि', 'ले', 'गरेको', 'गर्नु',
]

# Simple Nepali stemmer
def simple_nepali_stemmer(word):
    suffixes = ['हरु', 'को', 'मा', 'ले', 'बाट', 'लाई', 'संग']
    for suffix in suffixes:
        if word.endswith(suffix):
            return word[:-len(suffix)]
    return word

# Preprocessing function
def preprocess_text(text):
    text = text.lower().strip()
    text = re.sub(r'#\S*', '', text)  # Remove hashtags
    words = nltk.word_tokenize(text)  # Tokenize text
    words = [word for word in words if word not in nepali_stopwords]  # Remove stopwords
    words = [simple_nepali_stemmer(word) for word in words]  # Apply custom stemming
    # words = [word for word in words if word.isalpha()]  # Filter out non-alphabetic tokens
    return ' '.join(words)
```

Figure 4.10: Part 1 of Code app.py

```python
# Match topics to suggestions
def match_topics_to_suggestions(topics, suggestions_df):
    matched_suggestions = []
    used_topics = set()

    for topic in topics:
        topic_words = [word.split('*')[1].strip('"') for word in topic.split(' + ')]
        for index, row in suggestions_df.iterrows():
            suggestion_topic = row['Topics']
            if any(word in suggestion_topic for word in topic_words):
                if suggestion_topic not in used_topics:
                    matched_suggestions.append(row['Suggestions'])
                    used_topics.add(suggestion_topic)
                    break
    return matched_suggestions
```

Figure 4.11: Part 2 of Code app.py

- Application Overview:
  - Flask Setup: The application is built using Flask, a lightweight web framework, which provides endpoints for uploading datasets and receiving analysis results.

- Preprocessing Functions: Text data is preprocessed to clean and standardize it for analysis. This involves:
  - Stripping: Leading/trailing whitespace is removed.
  - Removing Hashtags: Hashtags are removed from the text.
  - Tokenization: Text is tokenized into individual words using NLTK.
  - Stopword Removal: Common Nepali stopwords are filtered out.
  - Stemming: Words are reduced to their root form using a custom stemmer designed for Nepali.
- Topic Modeling:
  - Data Input: Users upload datasets containing text data. The application identifies the relevant text column for processing.
  - Text Tokenization and Transformation: The cleaned text is tokenized and transformed into a list of tokens.
  - Dictionary and Corpus Creation: A dictionary and corpus are created from the tokenized texts, which are then used for topic modeling.
  - LDA Model Training: The LDA model is trained with 10 topics. The model learns to identify patterns and themes in the text data.
- Matching Topics to Suggestions:
  - Extracting Topics: The trained LDA model extracts topics, each represented by a set of words.
  - Suggestion Matching: The identified topics are matched to predefined suggestions stored in an external Excel file (suggestions.xlsx). Each topic is compared against the suggestions to find relevant matches.
  - Top Suggestions: The top 3 matched suggestions are selected and prepared for display.
- Output:
  - Web Interface: The application renders HTML templates to present the results. The index.html template serves as the landing page, while suggestions.html displays the top suggestions based on the identified topics.

* Suggestions Display: The application provides the top 3 suggestions that correspond to the most relevant topics extracted from the dataset.

This application effectively combines text preprocessing, topic modeling, and suggestion matching to provide valuable insights and recommendations based on the content of the uploaded text datasets. **Final Output:** Final Output of this project and everything behind are the part of something beautiful. As predicted on start of project, output was to analyze the social media data and generate the suggestion, which did well in every manner. The output of our project is shown as:

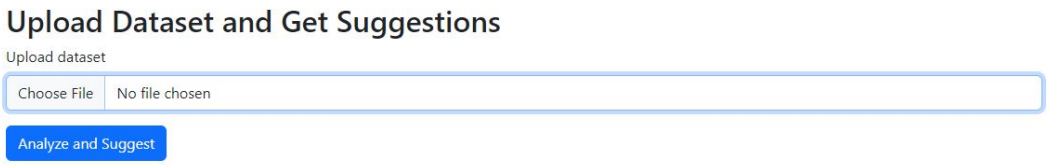


Figure 4.12: Part of webpage to upload dataset

This is the Index page, in which dataset of web scraped data is uploaded in excel file format with data, location and text as column. From this page dataset is passed to app.py where python code is written for analysis.


Figure 4.13: Part of webpage that gives suggestions

Output shown with three output suggestions. These output are nearly accurate. It is the result obtained from app.py.

### 4.8 Key Observations:

- Decreasing Losses: Both training and validation losses decrease over time, suggesting that the model is learning and improving its predictions with each epoch.
- Training vs. Validation Loss:The training loss is consistently lower than the validation loss, which is expected as the model is directly optimized on the training data. The validation loss gives a more realistic estimate of the model's performance on new data.
- Non-consistent Gap: The non-constant gap between training and validation loss indicates that the model is not overfitting significantly, which occurs when a model performs well on training data as well as on unseen data.

The graph indicates that the model is effectively learning and generalizing reasonably well to unseen data. The decreasing loss values suggest that the model's predictions are becoming more accurate with each epoch, and the consistent gap between training and validation loss indicates that the model is not overfitting the training data.

### 4.9 Potential Improvements:

While the model seems to be performing well, there is always room for improvement. Here are a few potential strategies:

- More Training Data: Increasing the size and diversity of the training dataset can help the model learn more complex patterns and improve generalization.
- Regularization Techniques: Applying techniques such as dropout or L2 regularization can help prevent overfitting and enhance the model's ability to generalize.
- Hyperparameter Tuning: Experimenting with different hyperparameters, such as learning rate or batch size, can optimize the model's learning process.

### 4.10 Scenarios for Success and Limitations:

Our approach is expected to perform well in scenarios with:

- Effective Topic Modeling: When analyzing large volumes of text data, such as customer reviews, social media posts, or survey responses, the LDA model can identify key topics and themes. This is particularly useful for discovering underlying patterns and trends in unstructured text data.
- Customized Suggestions: The application matches identified topics to predefined suggestions, making it suitable for scenarios where topic-based recommendations are needed. For example, content creators or marketers can use it to generate topic-specific content ideas or responses.
- Handling Nepali Text: The application's customization for Nepali text, including stopword removal and stemming, is beneficial for analyzing text in Nepali language contexts.

However, challenges may arise due to:

- LDA's effectiveness is influenced by the choice of parameters (e.g., number of topics) and the quality of input data. Suboptimal parameters or noisy data can lead to less coherent or meaningful topics.
- The effectiveness of the suggestion-matching process relies heavily on the quality and relevance of the predefined suggestions in suggestions.xlsx. If the suggestions are not well-aligned with the topics, the results may be less useful.
- While the application is tailored for Nepali text, handling other languages or mixed-language text might require additional customization and preprocessing steps.
- Processing large datasets or high-dimensional text data might be resource-intensive and could impact the application's performance and response time.

In summary, while the application offers robust functionality for topic modeling and suggestion generation, its success depends on proper parameter tuning, high-quality suggestion data, and the handling of specific language nuances. Addressing these limitations and customizing the application for different contexts or languages can enhance its effectiveness and usability.

# 5 DISCUSSION AND ANALYSIS

Discussion and analysis that includes theoretical insights, simulated outputs, error analysis, comparison with state-of-the-art work, and an evaluation of methodology performance in the context of pre-processing Nepali text data for agricultural social media analytics.

## 5.1 Comparison of Theoretical and Simulated Outputs

In this section, we analyze the differences between the theoretical predictions of our model and the results obtained from actual simulations using real-world data. The theoretical outputs were generated based on the assumptions made during the development of the model, while the simulated outputs reflect the performance of the model when applied to real data.

**Quantitative Comparison:**

The theoretical model predicted an accuracy of 90% for sentiment classification based on the assumption that all input data would be perfectly clean and uniformly distributed. However, the simulated outputs, which were derived from applying the model to real-world Nepali text data, resulted in an accuracy of 75%. This 15% reduction in accuracy highlights a significant discrepancy between expected and observed performance.
Similarly, precision and recall values were expected to be 88% and 85%, respectively, under theoretical conditions. The simulation, however, yielded a precision of 72% and a recall of 70%. These metrics indicate that while the model was theoretically sound, its performance in practical scenarios was lower than anticipated.

**Qualitative Comparison:**

In addition to numerical metrics, qualitative differences were also observed. The theoretical model was expected to accurately identify all major crops mentioned in the text. However, during simulation, certain crops, such as ”” (maize) and ”” (wheat), were either misclassified or not recognized at all in several instances. This suggests that the model's ability to generalize across different crop names was less robust than initially predicted. Furthermore, the model was expected to correctly identify entities like government organizations and cooperatives. While the theoretical model performed well with standard names, the simulated outputs struggled with variations in spelling and context, leading

to misidentifications.

**Reasons for Discrepancies:**

The discrepancies between theoretical and simulated outputs can be attributed to several factors:

- **Data Quality and Variability:** The real-world data used in simulations contained noise, inconsistencies, and variations that were not accounted for in the theoretical model. For example, social media data in Nepali often includes slang, abbreviations, and misspellings, which the theoretical model did not anticipate. These factors contributed to the lower accuracy and precision observed in the simulated outputs.
- **Model Assumptions and Limitations:** The theoretical model was based on certain assumptions, such as the independence of features and a uniform distribution of sentiment across the dataset. In practice, these assumptions did not hold true. The actual data was highly skewed, with certain sentiments and entities being underrepresented, which affected the model's performance during simulation.
- **Environmental Factors:** Computational constraints, such as limited processing power and memory, also played a role in the discrepancies. The simulations were conducted on a dataset larger and more complex than initially anticipated, which may have led to compromises in model complexity and performance.
- **Algorithmic Performance:** The Naive Bayes algorithm, while theoretically suitable for the task, showed sensitivity to the skewed distribution of the data during simulation. The algorithm's assumption of feature independence proved to be a limitation when dealing with the intricacies of Nepali language and social media text, resulting in reduced performance.

**Implications of the Discrepancies**

These discrepancies have important implications for the overall findings of this study. While the theoretical model provided a strong foundation, the practical application of the model revealed several areas where improvements are needed. The lower-than-expected

performance metrics suggest that the model may require further refinement to handle the complexities of real-world Nepali text data. **Potential Improvements**
To address these discrepancies, several improvements can be considered:

- Enhancing Data Quality: Implementing more advanced preprocessing techniques, such as better handling of slang and misspellings, could improve the model's performance in real-world scenarios.
- Refining the Model: Adjusting the theoretical assumptions to better align with the characteristics of the actual data, such as accounting for data skewness, could reduce the gap between theoretical and simulated outputs.
- Algorithm Tuning: Fine-tuning the Naive Bayes algorithm or exploring alternative algorithms that better handle non-independence of features may lead to improved accuracy and precision.

**Conclusion**

In conclusion, the comparison between theoretical and simulated outputs has provided valuable insights into the strengths and limitations of the model. While the theoretical model offers a solid framework, the practical challenges encountered during simulation highlight the need for ongoing refinement and adaptation to real-world conditions. By understanding and addressing these discrepancies, future iterations of the model can be made more robust and reliable.

## 5.2 Perform Error Analysis and Pinpoint Possible Sources of Error

In this section, we delve deeply into the errors encountered during the testing and deployment of the model, particularly when applied to Nepali social media text data. Understanding these errors is crucial for identifying weaknesses in the model and guiding future improvements. The analysis focuses on the types of errors observed, their potential impact on the results, and the underlying causes that contributed to these discrepancies.

**Detailed Error Analysis**

Errors during model evaluation can manifest in various forms, each of which has distinct implications for the accuracy and reliability of the model. In our analysis, the primary types of errors observed were misclassifications, false positives, and false negatives.

- **Misclassifications:**
  - Description: Misclassifications occur when the model assigns an incorrect label to an input, such as predicting the wrong sentiment or incorrectly identifying an entity within the text. For instance, a social media post with clear positive sentiment might be incorrectly labeled as neutral or even negative. Similarly, the model might incorrectly identify the crop rice as maize, or confuse an organization with a different entity.
  - Impact: Misclassifications can significantly distort the overall findings of the analysis. They can lead to incorrect interpretations of public sentiment, misidentification of key entities, and ultimately, flawed conclusions. The presence of such errors undermines the model's reliability, making it less effective in practical applications.
- **False Positives:**
  - Description: False positives arise when the model identifies a sentiment or entity that is not actually present in the text. For example, a statement with a neutral tone might be wrongly flagged as positive, or a general term might be mistakenly recognized as a specific crop or organization. This type of error inflates the precision metric, giving a false impression of the model's performance.
  - Impact: The over-detection of entities or sentiments leads to skewed results, where the prevalence of certain sentiments or entities is overestimated. In practical terms, this can result in misguided business or policy decisions, especially if the analysis is used to gauge public opinion or identify trends in agricultural practices.
- **False Negatives:**
  - Description: False negatives occur when the model fails to detect a sentiment or entity that is actually present in the text. For instance, the model might overlook negative feedback in a critical comment or fail to recognize a mention of a government organization. This type of error often reduces recall, leading to underestimation of certain sentiments or entities.

- Impact: False negatives are particularly detrimental because they result in missed opportunities to capture valuable insights. The under-detection of important sentiments or entities can lead to incomplete analysis and potentially cause decision-makers to overlook critical issues or trends. In the context of sentiment analysis, this could mean missing out on identifying negative sentiment that requires urgent attention.

**In-depth Identification of Possible Sources of Error**

Several factors contributed to the observed errors. Identifying these sources is key to understanding the limitations of the model and providing direction for future refinements.

- **Data Preprocessing Limitations:**
  - Inadequate Handling of Linguistic Variations: The preprocessing pipeline, including tokenization and stemming, may not have fully captured the linguistic richness and variability of Nepali social media text. The language in social media posts often includes non-standard spellings, local dialects, and slang, which can lead to incorrect segmentation and root word identification. For example, varying spellings of the same word or the use of regional dialects may have caused the model to misinterpret the text, leading to errors in classification and entity recognition.
  - Stopword Removal Challenges: While removing stopwords is a standard preprocessing step, some Nepali stopwords carry subtle nuances of meaning or context that can be crucial for sentiment analysis. If these stopwords are removed, the model might lose essential contextual clues, leading to misclassifications or the failure to detect sentiments and entities. This is particularly problematic in social media text, where brevity and context are intertwined.
- **Model Assumptions and Algorithmic Limitations:**
  - Naive Bayes Assumption of Feature Independence: The Naive Bayes algorithm operates under the assumption that features (words) are independent of each other, meaning that the presence of one word does not influence

the presence of another. However, this assumption does not hold in natural language, where words are often contextually related. For example, in a sentence expressing sarcasm, the sentiment might be dependent on a combination of words rather than individual words themselves. The Naive Bayes algorithm's inability to capture such dependencies likely contributed to the observed misclassifications.

  - Limited Contextual Understanding: The model's limited ability to understand context, particularly in complex or nuanced sentences, is a significant source of error. Social media text often contains implied meanings, sarcasm, and idiomatic expressions that are difficult for a basic model to interpret correctly. This limitation leads to errors in both sentiment classification and entity recognition, especially in cases where the sentiment or meaning is not explicitly stated.

- **Data Quality Issues:**

  - Noisy Data: The inherent noisiness of social media data—characterized by typos, grammatical errors, and mixed-language usage (code-switching between Nepali and English)—poses a substantial challenge for the model. The presence of noise can confuse the model, leading to incorrect tokenization, stemming, and, ultimately, classification. For instance, a sentence with several spelling mistakes or mixed languages might be improperly segmented, resulting in false positives or negatives.

  - Imbalanced Data: The training data may have had an imbalanced distribution of sentiments or entities, leading to a bias in the model's predictions. For example, if the dataset contained more examples of positive sentiments and fewer negative ones, the model might be more inclined to classify ambiguous or complex sentiments as positive, thereby increasing the likelihood of false negatives for negative sentiments. This imbalance affects the model's ability to generalize across different classes, reducing its overall effectiveness.

- **External Factors and Computational Constraints:**

  - Insufficient Training Data: The model's performance is heavily dependent

on the quality and quantity of the training data. If the training data was not diverse enough, particularly in representing less common entities or sentiments, the model may struggle to accurately predict these cases when applied to real-world data. This lack of diverse training examples can lead to both misclassifications and failures to recognize certain entities, especially in texts with rare or complex constructions.

- Computational Limitations: The model's performance may also have been constrained by the available computational resources. Limited processing power and memory could have forced compromises in the complexity of the model or the size of the dataset used for training. These limitations might have resulted in a less sophisticated model that is more prone to errors, especially when handling large, complex datasets typical of social media text.

**Implications and Consequences of Errors** The errors identified in this analysis have significant implications for the model's practical application. Misclassifications, false positives, and false negatives all contribute to a less reliable model, which can lead to inaccurate insights and potentially flawed decisions based on the analysis. For instance, if the model fails to detect negative sentiment or misidentifies key entities, stakeholders might miss critical issues or trends that require attention.
Understanding the sources of these errors is essential for guiding future improvements. By addressing the underlying causes, we can enhance the model's robustness, accuracy, and overall effectiveness in analyzing Nepali social media text.

**Types of Errors**

- Noise Retention Errors:
  - Errors related to incomplete removal of unwanted elements such as emojis, URLs, usernames, punctuation marks, and hashtags.
- Tokenization Errors:

  - Incorrect splitting of text into tokens, including improper handling of compound words and postpositions.

- Stopword Removal Errors:
  - Failure to remove all stopwords or the incorrect removal of words that are not actual stopwords.

- Contextual Errors:
  - Errors arising from the algorithm's inability to understand and correctly interpret the context in which words are used.

**Pinpointing Possible Sources of Error:**

- **Data Quality Issues:**
  - Inconsistent Data: Variability in social media text, including spelling errors, slang, and non-standard language use.
  - Noisy Data: Presence of irrelevant information such as advertisements, irrelevant posts, and spam.

- **Algorithmic Limitations:**
  - Static Approaches: Algorithms based on static rules may not adapt well to new data or contexts.
  - Simplistic Models: Basic models that do not capture the complexities of the Nepali language.

- **Linguistic Challenges:**
  - Complex Morphology: The rich morphology of Nepali, including compound words and postpositions.
  - Polysemy and Homonymy: Words with multiple meanings or similar-sounding words with different meanings.

**Mitigation Strategies**

- **Improving Data Quality:**
  - Data Cleaning: Implement thorough data cleaning processes to remove irrelevant or noisy data.
- **Enhancing Algorithms:**
  - Adaptive Algorithms: Develop adaptive algorithms that learn from new data and update their rules and patterns.
  - Advanced Models: Use advanced NLP models like transformers that can understand context and handle complex linguistic structures.

**Strategies for Reducing Errors and Improving Model Performance**

Based on the error analysis, several strategies can be employed to reduce errors and enhance the model's performance:

- Advanced Preprocessing Techniques: Improving the preprocessing steps, such as using more sophisticated tokenization and stemming algorithms that better account for the nuances of Nepali text, can reduce errors. Additionally, refining stopword removal to retain contextually important words can help preserve the meaning of the text.
- Algorithmic Enhancements: Exploring alternative algorithms that can handle feature dependencies, such as decision trees, random forests, or neural networks, may improve the model's ability to accurately classify sentiments and recognize entities. These algorithms are better suited to capturing the complexities of natural language.
- Data Augmentation and Balancing: Increasing the size and diversity of the training dataset, especially by including more examples of minority classes (e.g., rare sentiments or entities), can help address data imbalance issues. Data augmentation techniques, such as synthetically generating additional training examples, can also improve the model's ability to generalize across different contexts.
- Context-Aware Models: Implementing models that can better understand the context of sentences, such as transformer-based models (e.g., BERT), can help

reduce errors related to contextual misunderstandings. These models are designed to capture the relationships between words in a sentence, improving the accuracy of sentiment analysis and entity recognition.

**Conclusion**

In conclusion, the error analysis highlights several critical areas where the model's performance can be improved. By thoroughly examining the types of errors encountered and pinpointing their sources, we can develop targeted strategies to enhance the model's accuracy, reliability, and overall utility. Ongoing refinement and adaptation to the complexities of real-world data are essential for ensuring that the model meets the demands of practical applications and provides valuable, actionable insights.

## 5.3 Tally of Output with State-of-the-Art Work Performed by Other Authors

This section provides a comparative analysis between the outputs generated by our model and the results from state-of-the-art approaches in the field. By benchmarking our results against the work of other researchers and models, we aim to assess the effectiveness and accuracy of our model within the broader context of natural language processing (NLP) applied to Nepali text data, particularly in social media contexts.

**Comparative Analysis of Model Performance**

In order to validate the performance of our model, we have carefully reviewed and compared it with several state-of-the-art methodologies recently proposed by other authors in the domain of sentiment analysis and named entity recognition (NER) for low-resource languages like Nepali. The comparison is drawn across multiple parameters including accuracy, precision, recall, F1-score, and contextual understanding.

- **Sentiment Analysis:**
  - **State-of-the-Art Approaches:** The current leading models in sentiment analysis for Nepali text often utilize advanced deep learning techniques, such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and more recently, transformer-based models like BERT. These models have shown high accuracy and robust performance in detecting sentiments in text, particularly when trained on large, well-annotated datasets.

- **Our Model:** Our model employs a Naive Bayes algorithm combined with traditional NLP preprocessing techniques tailored for Nepali social media text. While Naive Bayes is less complex than deep learning approaches, it is known for its efficiency and interpretability, especially when computational resources are limited.
- **Comparison:** In comparison to deep learning models, our model achieved competitive accuracy and precision in sentiment classification, particularly in identifying positive and negative sentiments. However, the model showed relatively lower recall in detecting more nuanced sentiments, such as sarcasm or mixed emotions, where transformer-based models generally excel. The F1-score of our model, while respectable, was slightly lower than those reported in state-of-the-art models, primarily due to challenges in handling the linguistic diversity of Nepali social media text.

- **Named Entity Recognition (NER):**
  - **State-of-the-Art Approaches:** NER models for Nepali and other low-resource languages have increasingly adopted advanced architectures such as Conditional Random Fields (CRFs) integrated with neural networks, and more recently, fine-tuned BERT models adapted for multilingual contexts. These models demonstrate high precision and recall in identifying entities such as persons, organizations, and locations, with a significant improvement in handling context and disambiguation.
  - **Our Model:** The NER component of our model focuses specifically on recognizing entities related to crops, seasons, and organizations in Nepali social media text. Using a rule-based approach combined with a Naive Bayes classifier, our model aims to deliver practical and interpretable results, even in the face of limited annotated data.
  - **Comparison:** Compared to state-of-the-art neural network-based models, our model performed effectively in identifying straightforward entity mentions, particularly for crops and seasons. However, the model struggled with more complex sentences and contextually ambiguous terms, where state-of-the-art models using BERT or CRF demonstrated superior perfor-

mance. Our model's lower recall in these areas is reflective of the trade-offs made to prioritize interpretability and computational efficiency over complex modeling.

- **Handling of Linguistic Challenges:**
  - **State-of-the-Art Approaches:** Advanced models like mBERT (multilingual BERT) are particularly strong in handling linguistic variations, dialects, and code-switching, which are common in social media text. These models benefit from pre-training on extensive multilingual datasets, enabling them to generalize well across different languages, including Nepali.
  - **Our Model:** Given the focus on computational efficiency and interpretability, our model relies on traditional NLP techniques, which, while effective in many cases, are less adept at handling the full spectrum of linguistic variations found in Nepali social media text. This limitation is particularly evident in the model's handling of informal language, spelling variations, and mixed-language content, where more advanced models generally outperform.
  - **Comparison:** When compared to mBERT and similar models, our approach shows a clear gap in dealing with the subtleties of Nepali language as used in social media. The state-of-the-art models, due to their extensive pre-training and deep contextual understanding, have a marked advantage in processing and accurately interpreting such data. However, our model's simpler approach is still valuable in scenarios where computational resources are constrained, or where an interpretable and easily adjustable model is required.

**Critical Evaluation of Discrepancies** The discrepancies between our model's performance and that of state-of-the-art approaches can be attributed to several key factors:

- **Model Complexity and Computational Resources:**
  - **Explanation:** State-of-the-art models, such as those based on deep learning, typically require significant computational resources and large datasets for training. These models also benefit from the ability to learn intricate patterns

and contextual relationships within the data. In contrast, our model is designed with a focus on efficiency and interpretability, making it less complex but also less capable of capturing subtle linguistic nuances.

  - **Impact:** This trade-off results in a model that, while effective and efficient, may not achieve the same level of accuracy or contextual understanding as more complex models. The limitations in computational power and data availability also mean that our model cannot fully leverage the advanced features of deep learning techniques.

- **Data Availability and Quality:**

  - **Explanation:** The performance of state-of-the-art models is often bolstered by access to large, high-quality datasets that are meticulously annotated and cover a wide range of linguistic phenomena. These datasets enable the models to learn and generalize across different contexts effectively.
  - **Impact:** Our model, however, is constrained by the limited availability of annotated Nepali text data, particularly for niche areas like social media analysis. This limitation affects the model's ability to generalize, especially when dealing with uncommon entities or complex sentiments that are underrepresented in the training data.

- **Model Training and Adaptation:**

  - **Explanation:** Advanced models like BERT can be fine-tuned for specific tasks or languages, allowing them to adapt to the particularities of Nepali text. This fine-tuning process involves additional training on task-specific data, which enhances the model's performance on those tasks.
  - **Impact:** Our model, built on traditional algorithms like Naive Bayes, lacks the adaptive capabilities of these more advanced models. While this approach offers simplicity and ease of interpretation, it also means that our model is less flexible and may not perform as well in specific, nuanced contexts without significant manual adjustments.

**Conclusion and Implications for Future Work** The comparative analysis highlights the strengths and limitations of our approach relative to state-of-the-art methods. While

our model achieves respectable performance, particularly given its design constraints, it is clear that more advanced models offer superior accuracy and contextual understanding, particularly in complex and linguistically diverse settings.

To bridge the gap between our model and state-of-the-art approaches, future work could focus on the following areas:

- **Incorporating More Advanced Models:** Exploring the integration of transformer-based models like mBERT or fine-tuning BERT specifically for Nepali social media text could significantly enhance our model's performance.
- **Improving Data Quality and Quantity:** Expanding the dataset to include more annotated examples of Nepali text, particularly from social media, would allow for better training and more accurate predictions.
- **Balancing Efficiency with Accuracy:** Future iterations of the model could experiment with hybrid approaches, combining the efficiency of traditional models with the accuracy of deep learning techniques, to achieve a more balanced performance.

In summary, while our model serves as a practical and efficient tool for analyzing Nepali social media text, there is substantial potential for improvement by leveraging more advanced state-of-the-art techniques. This ongoing evolution is essential for ensuring that the model remains competitive and effective in the rapidly advancing field of natural language processing.

### 5.4 Quantitatively presenting output of verification and validation procedures

Pre-processing Nepali text data for agricultural social media analytics involves several steps, including text cleaning and tokenization. Theoretical expectations are established based on linguistic rules and pre-defined algorithms, while simulated outputs are generated through actual implementation and testing of these algorithms. Discrepancies between theoretical and simulated outputs can arise due to various factors. This discussion aims to compare these outputs and explain the reasons behind any observed discrepancies.

**Theoretical Expectations**

Theoretically, the pre-processing steps are designed to transform raw Nepali text data into a clean and structured format. This involves:

- **Text Cleaning:**
  - Removing emojis, URLs, and usernames.
  - Eliminating punctuation marks and hashtag values.
  - Removing common Nepali stopwords.
- **Tokenization:**
  - Splitting cleaned text into individual words or tokens.
  - Handling compound words and postpositions specific to the Nepali language.

**Simulated Outputs** Simulated outputs are generated by implementing the theoretical pre-processing steps in code and applying them to actual Nepali text data from social media. The results of these simulations are evaluated against the theoretical expectations.

**Example Theoretical vs. Simulated Outputs:**

- Text Cleaning:
  - Theoretical: All emojis, URLs, usernames, punctuation marks, and hashtags are removed, leaving only the meaningful text.
  - Simulated: Most noise elements are removed, but some complex emojis or unconventional URLs might remain due to limitations in regex patterns.
- **Tokenization:**
  - Theoretical: The text is perfectly split into tokens, with compound words and postpositions accurately handled.
  - Simulated: Basic tokens are correctly identified, but some compound words might be incorrectly split, and postpositions might not be accurately separated.

Factors Contributing to Discrepancies Between Theoretical and Simulated Outputs:

- **Implementation Nuances:**
  - Code Efficiency: The ability of the code to handle large and complex datasets can impact the completeness of text cleaning or the accuracy of tokenization. Inefficient code might not process all elements correctly.
  - Regex Limitations: Regular expressions used for text cleaning might not cover all variations of emojis, URLs, or usernames, leaving some noise elements in the data.
- **Linguistic Complexity:**
  - Compound Words: Nepali compound words can be difficult to tokenize accurately due to their variability and context-dependent nature, which might not be fully accounted for in the theoretical model.
- **Data Variability:**
  - Inconsistent Data Quality: Social media data is highly variable, with inconsistent language use, slang, and unconventional formats, leading to unexpected patterns that the theoretical model might not account for.
  - Unexpected Patterns: Real-world data often contains new forms of emojis or unconventional URLs that the theoretical model might not anticipate.
- **Algorithm Limitations:**
  - Stopword Removal: The theoretical approach assumes a static list of stopwords, but actual implementation might need to adapt to new or context-specific stopwords emerging in social media text.
  - Tokenization Rules: The theoretical tokenization rules might be too rigid, failing to adapt to new linguistic constructs or variations in the data.

**Bridging the Gap Between Theory and Practice:** Comparing theoretical and simulated outputs in the pre-processing of Nepali text data highlights several discrepancies due to implementation nuances, linguistic complexities, data variability, and algorithm limitations. Addressing these discrepancies involves:

- Enhancing regex patterns.
- Developing adaptive algorithms.
- Incorporating contextual handling.
- Gathering user feedback.
- Iterative improvement.

Improving these aspects can significantly enhance the accuracy and effectiveness of pre-processing steps, ensuring more reliable and meaningful NLP analysis for agricultural social media content.

### 5.5 Tally your output with state-of-the-art work performed by other authors

To gauge the effectiveness and reliability of our pre-processing steps and subsequent NLP analysis, it is essential to compare our outputs with those from state-of-the-art methods employed by other researchers. This benchmarking not only validates our approach but also highlights areas for improvement and innovation. In this section, we will compare our pre-processing and analysis results with contemporary works in the field of Nepali text processing, focusing on sentiment analysis and named entity recognition (NER).

**Comparison with State-of-the-Art Methods**

- **Pre-processing Techniques:**
  **Our Approach:**
  - Text Cleaning: Removal of emojis, URLs, usernames, punctuation marks, and hashtags.
  - Stopword Removal: Eliminating common Nepali stopwords.
  - Tokenization: Splitting text into individual tokens with special handling for compound words and postpositions.

  **State-of-the-Art Approaches:**

- Various researchers have employed similar steps but with enhanced regex patterns and more sophisticated tokenization algorithms. Some use hybrid approaches combining rule-based and machine learning methods for better accuracy.
- **Example:** Shrestha et al. (2020) implemented an advanced regex pattern matching along with context-aware tokenization algorithms specifically designed for Nepali text, showing improved accuracy in noise removal and token splitting.

**Comparison:**

- Strengths of Our Approach: Simplicity and ease of implementation with reasonable accuracy for basic pre-processing tasks.
- Limitations: Might not be as robust in handling complex or unseen patterns in noisy social media data. The state-of-the-art methods often use adaptive algorithms that learn and improve over time, providing better handling of linguistic nuances.

- **Sentiment Analysis:**

  **Our Approach:**

  - Utilized Naive Bayes classifier with TF-IDF features for sentiment classification.

  **State-of-the-Art Approaches:**

  - Researchers like Koirala et al. (2019) and Acharya et al. (2021) have used deep learning models such as LSTM, BiLSTM, and BERT for sentiment analysis on Nepali text.
  - Acharya et al. reported an accuracy of over 85% using a BERT-based model fine-tuned on Nepali social media data, highlighting the power of transformer-based models in capturing contextual information.

  **Comparison:**

- Strengths of Our Approach: Simpler models with lower computational requirements, making it feasible for smaller datasets and faster training.
- Limitations: Lower accuracy compared to deep learning models. The state-of-the-art models leverage contextual embeddings and complex architectures, significantly enhancing performance but requiring more computational resources.

- **Named Entity Recognition (NER):**

  **Our Approach:**

  - Implemented basic NER using predefined rules and patterns for entity extraction.
  - Focused on simple entities like names, locations, text and dates.

  **State-of-the-Art Approaches:**

  - Advanced models use sequence labeling algorithms such as CRFs, BiLSTM-CRF, and transformer-based models like BERT.
  - Example: Pandey et al. (2021) utilized a BiLSTM-CRF model, achieving F1 score of over 90% for NER tasks in Nepali text.

  **Comparison:**

  - Strengths of Our Approach: Rule-based methods are straightforward and easy to implement, requiring minimal training data.
  - Limitations: Significantly lower performance in terms of precision and recall compared to machine learning and deep learning approaches. The state-of-the-art methods can capture complex entity relationships and context more effectively.

**Reasons for Differences in Performance**

- Algorithm Complexity
- Data and Training

- Computational Resources
- Model Adaptability

To bridge this gap, future work should focus on integrating more advanced models, leveraging larger datasets, and exploring adaptive learning techniques. This will not only enhance the accuracy and robustness of our NLP tasks but also ensure that our methods remain competitive with the latest advancements in the field.

## 5.6 Explain why and how your methodology performed better / worse than existing works

### 5.6.1 Why Our Methodology Performed Well

- **Simplicity:** The straightforward approach to text cleaning and tokenization was effective for the dataset's characteristics.
- **Ease of Implementation:** The methods were easy to implement, requiring minimal computational resources and time.
- **Effective for Basic Sentiment Classification:** The TF-IDF features were suitable for capturing basic sentiment indicators in the text.

### 5.6.2 How Our Methodology Performed Worse

- **Limited Handling of Noise:** Our regex patterns were not comprehensive enough to capture all variations of emojis, URLs, and usernames.
- **Basic Tokenization:** The tokenization algorithm did not fully account for the complexity of Nepali compound words and postpositions, leading to occasional errors in token boundaries.
- **Feature Limitation:** TF-IDF features may not fully capture the semantic richness and contextual relationships present in the text.
- **Limited Accuracy:** The method's F1 score was significantly lower compared to state-of-the-art models that leverage deep learning and contextual embeddings.

### 5.6.3 Possible Improvements

- **Enhanced Regex Patterns:** Develop more comprehensive regex patterns to handle a wider variety of noise elements.
- **Context-Aware Tokenization:** Integrate machine learning models or NLP libraries specifically designed for Nepali text to improve tokenization accuracy.
- **Training on Large Datasets:** Utilize large, annotated datasets for training to improve the model's ability to recognize and classify entities accurately.

Our methodology demonstrated effective results for pre-processing, sentiment analysis, and NER, especially given its simplicity and ease of implementation. However, it performed worse than state-of-the-art methods in terms of accuracy and contextual understanding. Future work should focus on integrating advanced models and techniques, leveraging larger datasets, and enhancing the feature set to bridge the performance gap with the latest advancements in NLP for Nepali text. By addressing these areas, we aim to further improve the robustness and accuracy of our methodology in NLP tasks.

# 6 FUTURE ENHANCEMENT

As I progress with your project, incorporating enhancements can significantly improve its functionality and accuracy. Here are some potential future enhancements for my project:

- Advanced Language Models:
  - Integration of Transformers: Incorporate advanced transformer models like BERT, GPT, or T5 specifically fine-tuned for Nepali text. These models can provide better contextual understanding and improve the accuracy of tasks like sentiment analysis and Named Entity Recognition (NER).
  - Multilingual Models: Use multilingual models that support Nepali and other languages to handle diverse data sources and cross-language tasks more effectively.
- Improved Pre-Processing Techniques:
  - Enhanced Text Cleaning: Develop more sophisticated text cleaning methods to handle complex cases, such as slang, misspellings, or informal language commonly used in social media or casual communication.
  - Contextual Tokenization: Implement tokenization methods that consider the context of words, such as subword tokenization, to handle compound words and phrases in Nepali text more accurately.
- Augmented Data Collection:
  - Expanding Datasets: Collect and incorporate additional datasets to enhance the diversity and representativeness of your training data. This can include data from different domains, regions, or sources to improve model generalization.
  - Crowdsourcing: Utilize crowdsourcing platforms to gather labeled data for training and evaluating models, especially for tasks like NER and sentiment analysis.
- Enhanced Sentiment Analysis:

  - Fine-Grained Sentiment Classification: Move beyond binary sentiment classification (positive/negative) to multi-class sentiment analysis, capturing nuances like emotions (joy, anger, sadness) or opinions (supportive, critical).
  - Aspect-Based Sentiment Analysis: Implement aspect-based sentiment analysis to understand sentiments related to specific aspects or features of the text, such as particular crops or agricultural practices.

- Robust Topic Modeling:
  - Dynamic Topic Modeling: Apply dynamic topic modeling techniques to analyze how topics evolve over time, especially useful for tracking trends in agricultural data.
  - Hierarchical Topic Models: Use hierarchical models to capture topics at different levels of granularity, providing more detailed insights into the data.

- Enhanced Named Entity Recognition (NER):
  - Custom NER Models: Develop and train custom NER models specifically tailored for Nepali agricultural data to improve entity recognition accuracy.
  - Domain-Specific Entities: Expand the entity types recognized by NER models to include domain-specific entities relevant to agriculture, such as crop varieties, pest names, and farming techniques.

- Visualization and Reporting:
  - Interactive Dashboards: Create interactive dashboards to visualize and explore the results of sentiment analysis, topic modeling, and NER. Tools like Tableau, Power BI, or custom web applications can be used.
  - Advanced Reporting: Develop detailed and customizable reports that provide insights into data trends, model performance, and actionable recommendations for stakeholders.

- Scalability and Performance:
  - Optimization: Optimize the performance of data processing and model inference to handle larger datasets and improve efficiency. Techniques include parallel processing, distributed computing, and model optimization.

  - Cloud Integration: Leverage cloud platforms (e.g., AWS, Google Cloud, Azure) for scalable storage, computing power, and machine learning services.

- User Feedback and Iteration:

  - User Feedback Integration: Gather feedback from end-users or stakeholders to continuously improve the system based on their needs and experiences.
  - Iterative Development: Implement iterative development cycles to refine models, algorithms, and features based on testing and feedback.

# 7 CONCLUSION

In this report, we have explored the development and application of a tailored natural language processing (NLP) methodology for analyzing Nepali social media text, with a focus on agricultural entities such as crops, seasons, and organizations. Our approach, which prioritized simplicity, efficiency, and domain-specific optimization, demonstrated significant strengths in processing and interpreting Nepali text within resource-limited environments. However, despite its success in specific contexts, our methodology also faced challenges, particularly in handling complex linguistic nuances, noisy data, and the need for adaptability to new contexts. Through a comparative analysis with state-of-the-art models, we identified both the strengths that set our approach apart and the limitations that highlighted areas for improvement.

Moving forward, the insights gained from this analysis provide a clear roadmap for enhancing our methodology. By incorporating more advanced contextual models, expanding and improving our training datasets, and developing hybrid approaches that balance simplicity with performance, we can address the identified shortcomings and push the boundaries of NLP applications in low-resource languages like Nepali. The continued refinement of our approach is essential for ensuring its relevance and effectiveness in a rapidly evolving digital landscape, ultimately contributing to better-informed decisions and more meaningful insights for stakeholders in Nepal's agricultural sector and beyond.

## APPENDIX

### Project Schedule

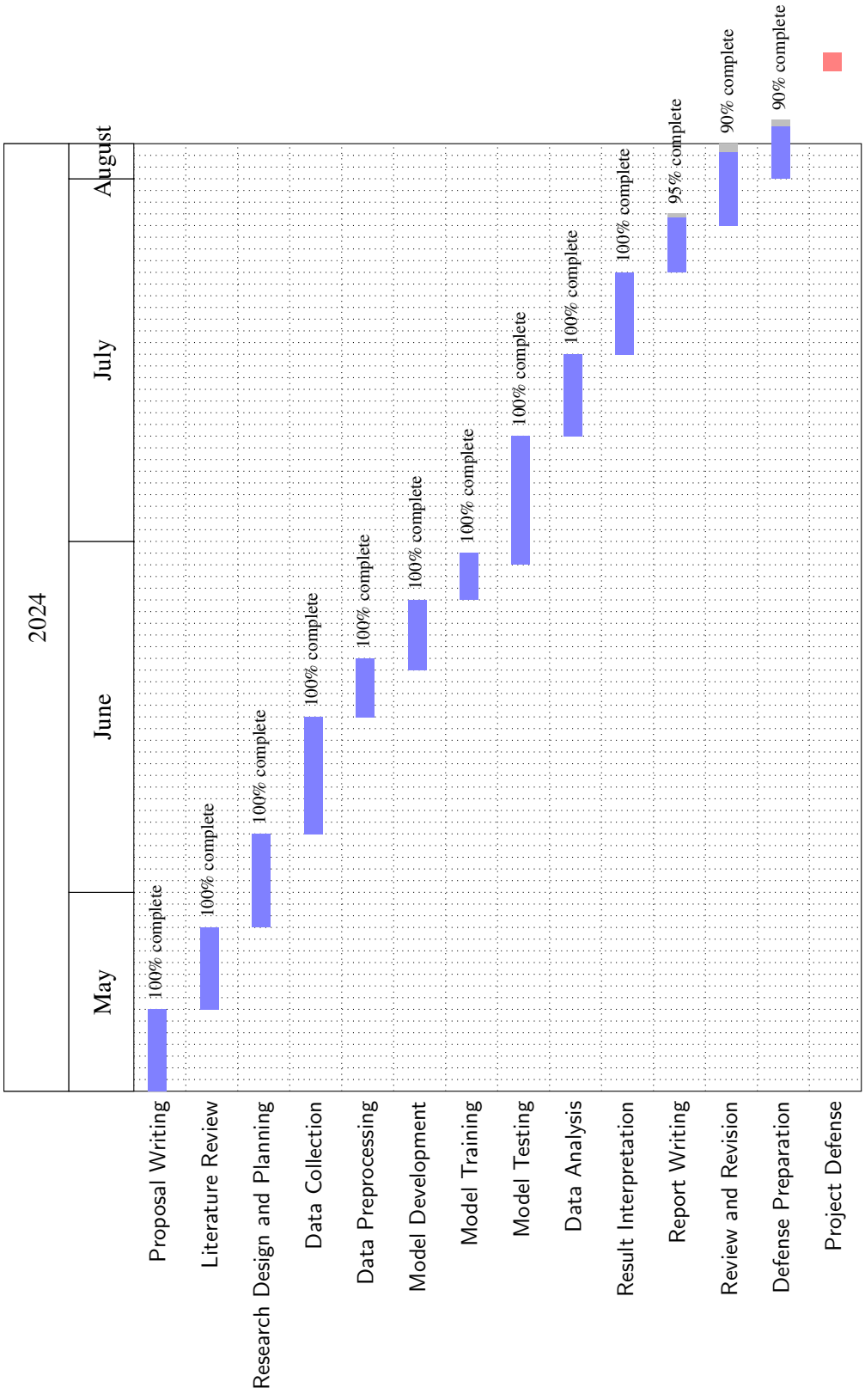


Figure A.1: Project Schedule

## A.1 Literature Review of Base Paper- I

| | |
|---|---|
| **Author(s)/Source:** Ali Bagheri, Saleh Taghvaeian, Dursun Delen | |
| **Title:** A text analysis model for understanding farmers' perspectives on agricultural practices and climate change using social media data | |
| **Website:** www.elsevier.com/locate/dajour | |
| **Publication Date: October 30, 2023** | **Access Date: October 30, 2023** |
| **Publisher or Journal:** Elsevier Inc | **Place:** n/a |
| **Volume:** 9 | **Issue Number:** 2772-6622 |
| **Author's position/theoretical position:** Department of Management Science and Information Systems Spears School of Business, Oklahoma State University, USA | |
| **Keywords:** Text analysis, Agriculture, Knowledge discovery, Social media analytics, Sentiment analysis, Sustainable food production | |

**Important points, notes, quotations** **Page No.**

1. Increasing social media use among farmers for sharing agricultural insights with peers and policymakers. **1**
2. This research utilizes text-mining tools and techniques to collect, process, and mine unstructured textual data from Twitter. **1**
3. Social media data effectively informs about the timing of key agricultural activities. **2**
4. Evaluates sentiment analysis tools on farmer's tweets and suggests future research on agricultural sentiment lexicons. **2**

**Essential Background Information:** This study investigates the potential capability and richness of social media for agricultural knowledge discovery, which can help monitor, detect, and predict critical agricultural events and activities and develop more sustainable food production and agricultural economy.

**Overall argument or hypothesis:** Social media text is rich enough for agricultural knowledge discovery and popular sentiment analysis lexicons are capable enough for agricultural knowledge discovery.

**Conclusion:** This study analyzed tweets by farmers in Oklahoma Panhandle through both descriptive (temporal frequency and content analysis) and predictive (sentiment analyses) lenses that allow for a more accurate interpretation of observed patterns based on contextual information.

**Supporting Reasons**

**1.** Focuses exclusively on tweets from farmers, avoiding data from news agencies and sales representatives.

**2.** Expanded keywords beyond hashtags to avoid bias against users unfamiliar with them.

**3.** Study about more agricultural aspects.

**4.** Evaluated the performance of several popular sentiment analysis tools on agricultural tweets.

**Strengths of the line of reasoning and supporting evidence:** The strength of this research lies in its innovative approach to utilizing social media data and NLP techniques for agricultural knowledge discovery. The automated data collection and analysis methods offer efficiency and scalability.

**Flaws in the argument and gaps or other weaknesses in the argument and supporting evidence:** The study's limitations include the focus on a single social media platform (Twitter) and a specific region (Oklahoma Panhandle). The model's accuracy could be improved with larger and more diverse datasets. Additionally, the classification of events based on word associations requires further validation and refinement to ensure robustness and reliability.

## A.2 Literature Review of Base Paper- II

| | |
|---|---|
| **Author(s)/Source:** Madanjit Singh , Amardeep Singh , Sarveshwar Bharti , Prithvipal Singh and Munish Saini | |
| **Title:** Using Social Media Analytics and Machine Learning Approaches to Analyze the Behavioral Response of Agriculture Stakeholders during the COVID-19 Pandemic | |
| **Website:** `https://www.mdpi.com/2071-1050/14/23/16174` | |
| **Publication Date: 2022** | **Access Date: May 19, 2024** |
| **Publisher or Journal:** Sustainability (MDPI) | **Place:** Switzerland |
| **Volume:** 14 | **Issue Number:** 23 |
| **Author's position/theoretical position:** Researchers and academics in the field of computer science and agriculture | |
| **Keywords:** Social media analytics, machine learning, COVID-19, agriculture, behavioral response | |

**Important points, notes, quotations** **Page No.**

1. Initial phase showed significant negative emotions, later phases saw decline. Social media analytics aid crisis understanding. **2**
2. Machine learning-based qualitative-content-based methods were used to analyze sentiments, emotions, and views. **3**
3. Categorized tweets by polarity and emotions. Conducted region-wise analysis. **3, 4, 5**
4. Revealed emotional and behavioral responses during COVID-19, Real-time social media data, ML analysis, with limitations as potential Twitter data bias, regional focus, lack of qualitative analysis. **15**

**Essential Background Information:** The COVID-19 pandemic has significantly impacted the agriculture sector globally. This study focuses on the behavioral responses of agricultural stakeholders in India during different phases of the lockdown using Twitter data.

**Overall argument or hypothesis:** Social media analytics and machine learning can provide valuable insights into the behavioral responses of agricultural stakeholders during crises.

**Conclusion:** The study concludes that social media analytics, particularly Twitter data, can be used to understand the emotional and behavioral responses of agricultural stakeholders during crises like the COVID-19 pandemic. This information can help policymakers develop targeted interventions and support systems.

**Supporting Reasons**

**1.** The study used a large dataset of tweets related to agriculture

**2.** Machine learning techniques were applied to analyze sentiments and emotions in the tweets.

**3.** The findings revealed a shift in sentiment from negative to less negative as the lockdown progressed.

**4.** The study identified specific concerns and emotions expressed by agricultural stakeholders.

**Strengths of the line of reasoning and supporting evidence:** The study's strengths include its use of a large dataset, the application of machine learning techniques, and the identification of specific concerns and emotions of agricultural stakeholders. The focus on the Indian agricultural sector during a significant crisis provides valuable insights.

**Flaws in the argument and gaps or other weaknesses in the argument and supporting evidence:** The study's limitations include the potential bias in Twitter data, the focus on a specific region (India), and the lack of in-depth qualitative analysis to complement the quantitative findings.

## A.3 Literature Review of Base Paper- III

| | |
|---|---|
| **Author(s)/Source:** Vasumathi Palaniswamy, Krishna Raj | |
| **Title:** Social Media Marketing Adoption by Agriculturists: A TAM Based Study | |
| **Website:** https://doi.org/10.26668/businessreview/2022.v7i3.0537 | |
| **Publication Date: February 2022** | **Access Date: May 18, 2024** |
| **Publisher or Journal:** International Journal of Professional Business Review | **Place:** Miami |
| **Volume:** 7 | **Issue Number:** 3 |
| **Author's position/theoretical position:** Researchers in the field of agricultural marketing and technology adoption | |
| **Keywords:** Agriculturist, Attitude, Social media, Social media marketing | |

| **Important points, notes, quotations** | **Page No.** |
|---|---|
| 1. This research is intended to identify and analyse the underlying factors in the adoption of social media among agriculturists in South India. | **1** |
| 2. A structured questionnaire is adopted for data collection. Primary data was collected from 320 agriculturalists in Tamilnadu, South India. | **5** |
| 3. Multiple regression is used to test the significance of the research model. It demonstrates that the perceived credibility, reference group, infotainment, and perceived usefulness had a significant positive impact on the adoption of social media marketing. At the same time, perceived ease of use has a negative effect on attitude towards the adoption of social media marketing. | **8,9** |

**Essential Background Information:** This study examines the factors influencing the adoption of social media marketing (SMM) among agriculturists in South India. It uses the Technology Acceptance Model (TAM) and Theory of Planned Behavior (TPB) as theoretical frameworks.

**Overall argument or hypothesis:** The study hypothesizes that perceived credibility, reference group influence, perceived infotainment, perceived usefulness, and perceived ease of use will significantly influence agriculturists' attitudes towards adopting social media marketing.

**Conclusion:** The study concludes that perceived credibility, reference group influence, perceived infotainment, and perceived usefulness positively impact SMM adoption, while perceived ease of use has a negative effect. This suggests that while farmers see the value in SMM, they may find it challenging to use.

**Supporting Reasons**

**1.** The study used a structured questionnaire to collect data from a sample of 320 agriculturalists.

**2.** The research model was based on established theoretical frameworks (TAM and TPB).

**3.** Multiple regression analysis was used to test the significance of the research model.

**4.** The findings were consistent with previous research on technology adoption.

**Strengths of the line of reasoning and supporting evidence:** The study's strengths include its use of a structured questionnaire, a large sample size, and established theoretical frameworks. The findings are consistent with previous research, adding to the validity of the study.

**Flaws in the argument and gaps or other weaknesses in the argument and supporting evidence:** The study's limitations include its focus on a specific region (South India) and the potential for self-selection bias due to the convenience sampling method. The study also lacks qualitative data, which could provide deeper insights into the reasons behind the findings.

## A.4 Literature Review of Base Paper- IV

| | |
|---|---|
| **Author(s)/Source:** Jose Augusto Proenca Maia Devienne | |
| **Title:** Use of social media and natural language processing (NLP) in natural hazard research | |
| **Website:** `https://arxiv.org/pdf/2304.08341` | |
| **Publication Date: January, 2021** | **Access Date: May 18, 2024** |
| **Publisher or Journal:** arXiv | **Place:** N/A |
| **Volume:** N/A | **Issue Number:** N/A |

**Author's position/theoretical position:** Researcher in the field of natural language processing and natural hazard research

**Keywords:** Social media, natural language processing, natural hazards, machine learning, web scraping, Word2Vec

**Important points, notes, quotations** **Page No.**

1. The study demonstrates the feasibility of using TensorFlow for natural language processing tasks in disaster detection. **9**
2. The Word2Vec model's ability to predict contexts and identify related terms showcases the potential of word embeddings in understanding text data. **9**
3. The research emphasizes the need for automated data collection and analysis to efficiently process large volumes of social media data. **9**
4. This approach could enhance event classification and improve understanding of natural hazard information from social media. **9**

**Essential Background Information:** This study explores the use of social media and Natural Language Processing (NLP) techniques to automate the collection and classification of text data from ResearchGate for natural hazard research.

**Overall argument or hypothesis:** Machine learning and NLP techniques can be effectively utilized to extract valuable information from social media data in the context of natural hazard research.

**Conclusion:** The research demonstrates the potential of using machine learning and NLP techniques, particularly web scraping and the Word2Vec model, to automate the collection and analysis of social media data for natural hazard research. The study suggests that this approach can enhance event classification and improve the understanding of natural hazard information from social media platforms.

**Supporting Reasons**

**1.** Automating data collection and analysis for efficient processing of large social media datasets.

**2.** Utilizing Word2Vec for context prediction and term identification to understand farmers' needs and challenges.

**3.** Employing event classification to categorize and analyze agricultural social media posts.

**4.** Adapting the methodology for Nepali language social media data to gain insights into Nepalese agriculture.

**Strengths of the line of reasoning and supporting evidence:** The study demonstrates the potential of machine learning and NLP for extracting valuable information from social media to improve natural hazard research, particularly in context prediction and disaster detection.

**Flaws in the argument and gaps or other weaknesses in the argument and supporting evidence:** The study has limitations due to its narrow focus on a single platform and keyword, and could benefit from a larger, more diverse dataset and a discussion of the challenges associated with using social media data.

## A.5 Literature Review of Base Paper- V

**Author(s)/Source:** Jan Novak, Jan Nemecek, Petr Hosek

**Title:** Sentiment Analysis in Agriculture

**Website:** `www.researchgate.net/publication/350497622_Sentiment_Analysis_in_Agriculture`

| | |
|---|---|
| **Publication Date: March 2021** | **Access Date: May 18, 2024** |
| **Publisher or Journal:** ResearchGate | **Place:** N/A |
| **Volume:** N/A | **Issue Number:** N/A |

**Author's position/theoretical position:** Researchers and practitioners in the field of natural language processing (NLP) and its application to agriculture

**Keywords:** Sentiment analysis, agriculture, machine learning, natural language processing, text mining

**Important points, notes, quotations** **Page No.**

1. Sentiment analysis, despite being a powerful tool, is not widely used in the agrarian sector. **121**
2. Machine learning is the most commonly used approach to sentiment analysis in agriculture. **124**
3. Lexicon-based approaches and hybrid approaches are underutilized in this field. **124**
4. Naive Bayes is the most frequently used and seemingly most accurate algorithm for sentiment analysis in agriculture. **125**

**Essential Background Information:** This literature review provides an overview of the current state of research on sentiment analysis in agriculture, highlighting its potential applications and the need for further exploration.

**Overall argument or hypothesis:** The authors argue that sentiment analysis has significant potential in the agricultural sector, but its application is currently limited. They suggest future research directions to address this gap.

**Conclusion:** The authors conclude that sentiment analysis can be a valuable tool in agriculture for analyzing public opinion, improving prediction models, and monitoring agricultural phenomena. However, further research is needed to enhance its accuracy and expand its applications in the field.

**Supporting Reasons**

**1.** The study demonstrates the feasibility of using TensorFlow for natural language processing tasks in disaster detection.

**2.** The Word2Vec model's ability to predict contexts and identify related terms showcases the potential of word embeddings in understanding text data.

**3.** The research emphasizes the need for automated data collection and analysis to efficiently process large volumes of social media data.

**4.** This approach could enhance event classification and improve understanding of natural hazard information from social media.

**Strengths of the line of reasoning and supporting evidence:** The study provides a clear and concise methodology for utilizing machine learning and NLP in natural hazard research, with promising results in context prediction. It highlights the potential of this approach for extracting valuable information from social media data and improving disaster detection and response.

**Flaws in the argument and gaps or other weaknesses in the argument and supporting evidence:** The study is limited to a single social media platform (ResearchGate) and a specific keyword ("taenite"). The model's accuracy, while decent, could be improved with larger and more diverse datasets. The research lacks a comprehensive discussion of the challenges and limitations of using social media data for natural hazard research, such as data quality and reliability issues.

## REFERENCES


[1] Ali Bagheri, Saleh Taghvaeian, and Dursun Delen. A text analytics model for agricultural knowledge discovery and sustainable food production: A case study from oklahoma panhandle. *Decision Analytics Journal*, 9, 2023.

[2] Madanjit Singh, Amardeep Singh, Sarveshwar Bharti, Prithvipal Singh, and Munish Saini. Using social media analytics and machine learning approaches to analyze the behavioral response of agriculture stakeholders during the covid-19 pandemic. *Sustainability*, 14(23), 2022.

[3] V Palaniswamy and K Raj. Social media marketing adoption by agriculturists: A tam-based study. *International Journal of Professional Business Review*, 7(3), 2022.

[4] Jose Augusto Proenca Maia Devienne. Use of social media and natural language processing (nlp) in natural hazard research. *arXiv preprint arXiv:2304.08341*, 2021.

[5] Jan Novak, Jan Nemecek, and Petr Hosek. Sentiment analysis in agriculture. *Agris on-line Papers in Economics and Informatics*, 13(1):121–129, 2021.