**CSCE 5290: Natural Language Processing**

**Project Proposal**

**Title: Multilingual Hate Speech Detection**

**Members**

Manoj – 10012 xxx

Dinesh – 1002xx

**Github Link:** https://github.com/manojpedarla/hatespeech-detection

**1. Motivation**

In today’s digital age, online platforms have become central spaces for communication and information. However, they are prone to be abused, with hate speech, offensive language and toxicity. This exposure can lead to significant psychological harm leading to mental health impact. Additionally hate speech and offensive language often encourage polarization within communities making it harder for people to engage in constructive dialogue and work towards common goals. This hate speech comes in multiple languages and detecting them a difficult task. So, this project aims to address the growing need for effective moderation by developing a multilingual hate speech detection system.

**2. Significance**

Traditional methods to filter out hate speech was to do a keyword-based filtering. In this approach, terms are scanned for keywords to flag them as potentially problematic content. This is not the best approach as context of the sentence matters more than word itself. For instance, “The movie I watched was sick” and “His brain is sick” have polar opposite tone. This is true for multiple languages as well. So in our project, we plan to create a classifier that considers the context rather than words in different languages. This will help moderate toxic content and help in creating more healthier online community.

**3. Objectives**

By the end of the project, our goal is to build a classification model that can classify hate speech from neutral or positive speech. This is achieved by

* **Preprocessing text data:** Handling uppercase and lower cases, removing special characters.
* **Removing stop words**
* **Text processing techniques:** Techniques like lemmatization, stemming, etc
* **Tokenization:** Serializing the words**,** embeddings
* **Model Selection:**  Using pre-trained models like mBERT or building our own models depending on the accuracy.
* **Training**
* **Evaluation:** Using metrics like precision, recall and F1-score to find how well the model performs

**4. Features**

* **Multilingual Detection:** Ability to detect the language
* **Hate speech filtering:** Classifying which sentence is hate speech and which is not
* **Context Awareness:** Recognize context to reduce misclassifications.

**5. Dataset**

The dataset will be sourced from Hate Speech Data (https:// hatespeechdata.com), a comprehensive repository of multilingual hate speech datasets collected from various online platforms.

**Details of the Dataset:**

* **Size:** The dataset contains millions of labeled samples in various languages such as English, Spanish, Arabic, and others. Each dataset has different size and percentage of abuse. Some containing as high has 21 M records and others as low as in thousands.
* **Type:** The dataset includes text data, with each entry labeled as hate speech, offensive speech, or neutral. Additional metadata may include the source of the data, the language, and user information (if applicable).
* **Sources:** Social media platforms, news comments, and public forums.
* **Preprocessing:** The preprocessing steps will include:
  + Combining datasets
  + Removing unwanted features
  + Dealing imbalance with oversampling or other techniques
  + Handling case sensitiveness and special characters
  + Stopword removal
  + Tokenization

**6. Visualization**

|  |
| --- |
| **Data Collection** |
|  |
| **Preprocessing** |
|  |
| **Tokenization** |
|  |
| **Model Building** |
|  |
| **Model Evalutaion** |
|  |

1. **Data Collection:** Gathering data from webite, labeling data with their respective language name, handling imbalace
2. **Preprocessing:** Handling case sensitiveness, Removing special characters, removing unwanted features, stopwords removal.
3. **Tokenization:** Lemmatization/Stemming, Creating embeddings using NLP Techniques like Word2Vec, BagOfWords
4. **Model Training:** Use pretrained models, Builiding owne models, Training on multiple epochs
5. **Model Evaluation:** Choosing right metrics, Visualizations