**CSCE 5290: Natural Language Processing**

**Project Proposal – Group 14**

**Title: Multilingual Hate Speech Detection**

**Members**

Manoj pedarla - 11713124

Dinesh kumar Patibandla – 11689319

Neeharika Gali – 11660457

Bhavana Nalabothu - 11646466

**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:** Removing un-important words from data to reduce noise.
* **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:** Feeding processed data into a model, adjusting model parameters, optimizing to differentiate hate speech while not overfitting
* **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**

OLID features a multilingual dataset with five languages. The languages included are:

* + Arabic
  + Danish
  + English
  + Greek
  + Turkish

The datasets come from various social media platforms like twitter. It can be found at <https://sites.google.com/site/offensevalsharedtask/olid>

**Details of the Dataset:**

* **Size:** The dataset contains around 50 thousand labeled training samples in various languages such as English, Spanish, Arabic, and others, of which Turkish has the highest number of records (31277 train and 3515 test).
* **Type:** The dataset includes text data, with each entry labeled as OFF or NOT (offensive and not offensive respectively).
* **Sources:** Social media platforms, news comments, and public forums.
  + <https://arxiv.org/pdf/2004.02192>
  + <https://arxiv.org/pdf/1908.04531>
  + <https://arxiv.org/pdf/2004.14454>
  + <https://arxiv.org/pdf/2003.07459>
  + https://aclanthology.org/2020.lrec-1.758/
* **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