CSE440: Natural Language Processing II

Lab Project

Multi-Class Text Classification: A Comparison of Word Representations and ML/NN Models

Task:

You will be assigned one of the following datasets: You must use only your assigned dataset throughout the project.

Data Access Link: 🗀 Project Dataset

Project Guidelines: **[READ THIS VERY CAREFULLY!]**

Begin by loading and exploring your assigned dataset. The dataset is already split into training and testing, provided as separate CSV files. Use the training set for model training and validation, and the testing set strictly for final evaluation.

In this project, you will do extensive Exploratory Data Analysis (EDA) to uncover meaningful insights from your dataset. Document your findings in both the report and the corresponding .ipynb notebook. After EDA, you will decide what pre-processing needs to be done in order to get clean text. Then, you need to apply necessary preprocessing (Stopward removal, Stemming/Lemmatization, etc.) based on your findings to generate clean data. Note that, what preprocessing needs to be done will be decided by you. Then, you will proceed to the experimental phase. You can experiment with different preprocessing techniques and compare which one gives you the best result. After that, you will implement various word representation techniques and apply different Machine Learning (ML) and Neural Network (NN) models. Specifically, you are required to implement all the word representation techniques covered in the labs and assignments, which include:

- TF-IDF (For ML and DNN Models)
- Skip-gram (For all NN models)

Note that, for skipgram, you can either train the representation yourself using the given dataset or you can download a pretrained one.

Next, you will train any **one** of the machine learning models below:

- Random Forest
- Logistic Regression
- Naive Bayes

Then, you will train **all** of the following neural network models below:

- Deep Neural Network
- SimpleRNN
- GRU
- LSTM
- Bidirectional SimpleRNN
- Bidirectional GRU
- Bidirectional LSTM

You are not required to implement Transformer-based or other cutting-edge models, but you are welcome to explore them if time permits.

For each model, you must manually tune the hyperparameters, using validation performance to guide your choices. You are required to experiment with:

- TF-IDF with one ML model and a deep neural network. [2 experiment]
- Skip-gram with all neural network models. [7 experiments]

You need to tune the hyper parameters while experimenting with the models.

Note that the experiments will require at least 9 separate training runs. Therefore, start your project early. Divide the work with your groupmates. You will not be able to finish it if you begin on the last day. Also, usage of AI is allowed as long as you understand what you are doing.

Evaluate the performance of each model (paired with the word representation techniques) using appropriate metrics. You must calculate and report the following:

- Accuracy
- F1-score (macro)
- Confusion matrix
- Classification report

Use both visual and tabular representations to compare model performance clearly and effectively.

At the end of your experiments, you must:

- Identify and highlight the best-performing and worst-performing models.
- Provide a detailed comparison between the ML model and the best-performing NN model.

- Your report should contain the reasons for all decisions made including choosing the hyper parameter of each model.

Deliverables:

Your final submission must include the following components:

Jupyter Notebook (.ipynb File)

- The notebook must be well-organized and clearly structured.
- Use appropriate Markdown cells for section headings, explanations, and analysis.
- Include well-commented code to ensure readability and clarity.
- All plots, tables, and evaluation results should be embedded within the notebook.
- Clearly indicate any hyperparameter tuning experiments and log the results accordingly.

Project Report (IEEE Double-Column Format, Up to 6 Pages – PDF)

Latex Word

- **Abstract:** A brief summary of your overall approach, techniques used, and key findings.
- **Introduction:** Background of the task, motivation for the project, and a brief overview of the dataset.
- **Methodology:** Description of your exploratory data analysis, preprocessing techniques, word representation methods, and model architectures.
- **Results:** Comparative analysis of the models using relevant evaluation metrics. Include tables and visualizations to support your discussion.
- **Conclusion:** Summarize the key takeaways from your experiments, note any limitations encountered, and suggest possible future improvements.
- **References:** Cite all research papers, tools, libraries, or resources used in your project. Use IEEE citation style.

Viva/Presentation:

- You will attend a short viva or presentation session where you will explain your approach, design decisions, and the results you obtained.
- Each team member must be prepared to answer questions about any aspect of the project, including the code and methodologies. It does not matter who did what, you must be able to answer all the questions.
- As the use of AI tools is permitted, we expect you to have a clear understanding of your implementation and be able to explain your work confidently.

Submission Instructions:

You must submit a compressed ZIP file containing the following:

- Your report (GroupNo_ID1_ID2_ID3.pdf)
- All code files or notebooks (GroupNo_ID1_ID2_ID3.ipnyb/py)

Mark Distribution

The final grading will be based on the following components:

- Viva/Presentation: 6 marks
- Code: 2 marks
- Report: 3 marks