Project 2024-2025

Sentiment Analysis for domain-specific texts

Project Description:

Sentiment Analysis is a critical natural language processing task that focusses on extracting and understanding the sentiments associated with a product or service mentioned in text, such as product reviews. The aim of this project is to guide you through the process of designing, implementing, and evaluating sentiment analysis models using various NLP techniques, including traditional machine learning, deep learning, and transfer learning models. During the project you will have the opportunity to explore the entire pipeline from data collection, preprocessing, model training, and evaluation, to fine-tuning advanced models such as transformers.

Objectives:

- Gain practical experience in sentiment analysis for domain-specific texts.
- Understand the challenges of data collection, labelling, and preprocessing.
- Explore various modelling approaches (e.g., classical machine learning models, deep learning models like LSTMs or transformers).
- Evaluate model performance and interpret the results.

Milestones and Structure:

1. Domain Selection and Dataset Creation: Gather or find a dataset of product reviews from various online sources (e.g., movie reviews, product reviews, social media posts). The dataset should contain reviews with sentiment labels (e.g., positive, negative, neutral).

Tasks:

- Select a specific domain where sentiment analysis is valuable.
- Collect a dataset (public sources such as IMDB, Amazon, or scraping specific websites) or find a pre-labeled dataset from platforms like Kaggle
- Label the data with sentiment categories (e.g., positive, neutral, negative) if no pre-labeled data is available.
- Explore the dataset: perform an initial exploratory data analysis (EDA), including class distribution, text length, and domain-specific challenges.

Milestone: A report outlining the chosen domain (e.g., movie reviews, financial reports, customer feedback) and a labeled dataset

2. Preprocessing and Text Analysis: Preprocess text data by tokenization, stop word removal, stemming/lemmatisation, and feature extraction.

Tasks:

- Perform text cleaning: remove noise, handle emojis, stop words, and other irrelevant tokens.
- Tokenize and vectorize the data using techniques like TF-IDF or word embeddings (e.g., Word2Vec, GloVe).
- Handle domain-specific linguistic challenges (e.g., jargon, abbreviations, or informal language).

Milestone: Cleaned and preprocessed dataset ready for modeling.

3. Baseline Model Development: Explore and/or develop sentiment analysis models that predict the sentiment (positive, negative, neutral) using machine learning techniques, which can classify the sentiment of the reviews.

Tasks:

- Implement classical machine learning models for sentiment classification (e.g., logistic regression, SVM, Naive Bayes).
- Train and validate the models using appropriate evaluation metrics such as accuracy, precision, recall, and F1 score.
- Discuss the strengths and limitations of classical methods on the given dataset.

Milestone: At least two working baseline sentiment classifiers (e.g., logistic regression, SVM).

4. Deep Learning Model Development: Explore deep learning models for classifying the sentiment of the reviews.

Tasks:

- Implement more advanced models using deep learning architectures (e.g., LSTMs, CNNs, or transformers like BERT).
- Fine-tune pre-trained models (such as BERT, DistilBERT) for the sentiment analysis task.
- Compare and contrast performance with the baseline classical models.

Milestone: At least two deep learning-based sentiment analysis models (e.g., LSTM, BERT).

5. Evaluation, Performance Analysis and Visualization: Implement evaluation metrics (e.g., F1-score, accuracy, precision, recall) to assess each model's performance on sentiment classification tasks. Use cross-validation and other relevant techniques to ensure robust and unbiased evaluation. Explore techniques for handling imbalanced datasets and achieving high model performance. Evaluate and compare different sentiment classification techniques.

Tasks:

- Evaluate the models using multiple metrics, including accuracy, precision, recall, F1 score, and confusion matrix.
- Conduct error analysis to understand the types of sentiment misclassifications.
- Compare the performance of classical and deep learning models on the domain-specific dataset.
- Investigate the impact of different preprocessing techniques
- Analyse specific cases where the model fails or makes ambiguous predictions.
- Present the results in an easily interpretable format, such as graphical visualisations. Create insightful visualizations to help users understand customer sentiments better.

Milestone: A comparative analysis report of different models.

6. Final Project Report and Presentation: Compile all findings into a final report and deliver a formal presentation

Tasks:

- Create comprehensive documentation, describing the problem and detailing the methodology, including data collection, preprocessing, model architecture, evaluation methodology, experimental results, challenges and recommendations. Finally conclude and suggest potential avenues for further research and improvements in sentiment analysis.

- Prepare a presentation to showcase the model, highlighting challenges, successes, and future work.

Milestone: A final report and a presentation summarizing the project findings.

Deliverables:

a) **The report** in a pdf format with the title of your project, your name/surname, and your student id number in the first page, and the references in the last page.
b) **The link** (ie GitHub, Kaggle, Colab etc) of your code and dataset used for your project.

Due date for eclass submission: ***…./02/2025***
Presentation date: ***…./02/2025***

Technologies and Tools:

- **Programming Languages:** Python
- **Libraries/Frameworks:**
  - NLTK, SpaCy (for preprocessing)
  - Matplotlib/Seaborn (for data visualization)
  - Scikit-learn (for baseline models)
  - Keras/TensorFlow or PyTorch (for deep learning models)
  - Hugging Face Transformers (for pre-trained models)

Grading Criteria

Criteria	Description	Weight (%)
Problem Understanding and Domain Relevance	Demonstration of a clear understanding of sentiment analysis and its unique challenges within the chosen domain - Clear explanation of the importance of sentiment analysis in the selected domain, highlighting domain-specific challenges (e.g., domain jargon, context, sensitive language).	**10%**
Dataset selection	Quality and relevance of the dataset, including domain-specific text. Completeness of sentiment annotations (positive, negative, neutral). - Appropriateness and quality of the dataset collected for the domain - Accuracy and thoroughness of manual or semi-automated sentiment annotations for the dataset. - Comprehensive exploratory data analysis (e.g., distribution of sentiment labels).	**15%**
Data Preprocessing	Effectiveness of preprocessing for domain-specific language, handling of jargon, and preparation of text for modelling. - Adequate cleaning and preprocessing of text, including removal of irrelevant elements, normalization of domain-specific terms, and text tokenization. Specific strategies to handle domain-specific challenges (e.g., abbreviations, compound terms, context-specific phrases).	**15%**

Model Selection and Justification	Selection and development of appropriate sentiment analysis models tailored to the domain's specific challenges - Justified selection of sentiment analysis models (e.g., SVM, LSTM, BERT), particularly those adapted for the domain's specific text characteristics. - Proper design of the model, including feature extraction and sentiment classification steps.	**15%**
Model Evaluation	Comprehensive evaluation of the model's performance using appropriate metrics (accuracy, F1 score, precision, recall), and its ability to handle domain-specific nuances. - Correct application of relevant evaluation metrics to assess sentiment classification performance. - Detailed analysis of misclassifications, particularly those related to domain-specific text.	**15%**
Final Report	Quality of the final report, including explanations of methods, results, and conclusions. - Clear and structured explanation of the methodology, results, and design choices in the final report. - Critical discussion of results through insightful discussion of the system's strengths, limitations, and areas for improvement in the context of domain-specific challenges.	**10%**
Final Presentation	Effectiveness of the project presentation, clarity of communication, and ability to explain the technical aspects of the project. - Well-structured presentation, covering all key aspects of the project in a logical flow. - Clear explanation of complex concepts, models, and results to both technical and non-technical audiences. - Ability to answer questions thoughtfully, demonstrating a deep understanding of the subject matter and the project's details.	**10%**