Project Plan

1. Set up the environment & install dependencies
2. Load and preprocess the dataset (Hugging Face's CoNLL-2003)
3. Use Word Embeddings (Pretrained GloVe)
4. Build the NER Model
5. Train the model
6. Evaluate results (Precision, Recall, F1-Score)
7. Save and document everything in a GitHub repository

Problem Definition (Explain what NER is)

Named Entity Recognition (NER) is a crucial subfield of Natural Language Processing (NLP) focused on identifying and classifying key elements within text into predefined categories. These categories typically include entities such as person names, organizations, locations, dates, and more.

How NER Works:

1. Entity Detection: The system scans the text to locate sequences of words that constitute entities.
2. Entity Classification: Once detected, these entities are categorized into specific classes like person names, organizations, or locations.

Applications of NER:

- Information Extraction: Transforming unstructured text into structured data by identifying entities such as company names or financial indicators, which is valuable in fields like business intelligence and academic research.
- Sentiment Analysis: Identifying brands, products, or services in customer feedback to understand opinions and market trends.
- Search Engine Optimization: Enhancing search algorithms by understanding user queries through identified entities, leading to more relevant search results.

Challenges in NER:

- Ambiguity: Words can have multiple meanings; for example, "Apple" could refer to a fruit or the technology company.
- Domain Adaptation: NER systems trained in one domain (like news articles) may not perform well in another (such as medical records), requiring domain-specific adjustments.

Advancements in machine learning, especially deep learning techniques, have significantly improved the accuracy and adaptability of NER systems across various applications.

Dataset CoNLL-2003 load from hugingface

Word Embeddings (GloVe) https://nlp.stanford.edu/projects/glove/

Model Architecture (BiLSTM)

Training Process & Optimization (Adam, Cross-Entropy Loss)

Results & Evaluation