**Week 11: Lab Programs**

**Question 1: Deep Learning**

**Q1.** Build a complete NLP pipeline to perform **sentiment classification** on Amazon product reviews using deep learning methods like **LSTM and GRU**, with **Word2Vec embeddings**.

Consider the dataset: **Amazon Product review for Cell Phone Accessories**, http://snap.stanford.edu/data/amazon/productGraph/categoryFiles/reviews\_Cell\_Phones\_and\_Accessories\_5.json.gz

1. **Data Loading and Preprocessing**
   * Load the JSON file and extract fields like reviewText and overall.
   * Convert the ratings into binary sentiment labels: 1 (Negative = 1 or 2), 0 (Positive = 4 or 5), discard 3.
   * Preprocess the text (tokenization, stopwords removal, lowercase conversion, etc.).
2. **Word2Vec Embedding**
   * Train a Word2Vec model using gensim on the tokenized text.
   * Generate an embedding matrix using Word2Vec vectors for use in the neural network.
   * Justify why Word2Vec is preferred over one-hot encoding for this task.
3. **LSTM Model Implementation**
   * Build an LSTM-based model using Keras with:
     + An embedding layer (initialized using Word2Vec)
     + LSTM layer (128 units)
     + Fully connected dense layer for binary classification
   * Train and evaluate the model using accuracy and confusion matrix.
4. **GRU Model Implementation and Comparison**
   * Replace the LSTM layer with a GRU layer and keep the rest of the architecture the same.
   * Train and evaluate the GRU model on the same data.
   * Compare the performance (accuracy, training time, etc.) of both models.
5. **Performance Visualization and Reporting**
   * Plot training/validation accuracy and loss for both models.
   * Generate and interpret confusion matrices.
   * Create word clouds for positive and negative reviews.
   * Submit an implementation file summarizing your findings and model comparison.

Q2. Explore the use of **BERT and transformer-based models** for two core NLP tasks:

* **Masked Language Modeling (Next Word Prediction)**
* **Language Translation (English to Hindi)**  
  using **open datasets** and the Hugging Face Transformers library.

**Dataset can be considered from the following information:**

1. **Next Word Prediction (Masked Language Modeling)**  
   Use any English sentence dataset such as:
   * Kaggle - BookCorpus Dataset (Subset)
   * Alternatively, use Hugging Face Dataset: bookcorpus or wikitext or IndicTrans (AI4Bharat)
2. **Language Translation (English-Hindi)**  
   Use the open dataset:
   * Kaggle - English to Hindi Translation Dataset
   * Alternatively: Hugging Face Dataset: opus\_books, flores, or iitb-english-hindi
3. **Task 1:** Next Word Prediction using BERT
   * Use the **Masked Language Modeling (MLM)** feature of bert-base-uncased.
   * Input a sentence with a masked word:  
     *"Artificial Intelligence is the [MASK] of the future."*
   * Use Hugging Face's pipeline('fill-mask') and display the top 3 predicted words.
   * Interpret the results and how BERT uses context.
4. Task 2: English-to-Hindi Translation using MarianMT or **IndicTrans2**
   * Load the pretrained model ai4bharat/indictrans2-en-hi using Hugging Face.
   * Translate 5 simple English sentences to Hindi.
   * Display the outputs and compare with human-translated equivalents.
   * Evaluate the translation quality using **BLEU score** or manual observation.