- **The most frequent sense:** This is the baseline for the given task, in the method we choose the first synset from the wordnet dataset, the most frequently used synset is first in wordnet dataset
- **Lesk algorithm:** For the Lesk algorithm I have used the nltk library and applied the Lesk algorithm on the given dataset.
    - **Preprocessing:** To enhance the dataset, preprocessing involves tokenization, lemmatization, and the removal of stop words from the context to eliminate unnecessary words.
    - **Results:**

Method	Dev set	Test set
Most frequent sense	65.46 %	62.75 %
Lesk Algorithm	31.95 %	29.51 %

Table 1: Baseline results, given results are in % Accuracy

- **Bootstrapping:**
    - **Dataset:**
      Five distinct lemmas (**"plan"**, **"climate"**, **"focus"**, **"term"**, and **"path"**) were selected for bootstrapping. Using ChatGPT, 18 diverse texts were generated for each lemma, encompassing three different sentiments for experimentation.
    - **Text preprocessing:** Text preprocessing involved converting capital letters to lowercase, sentence tokenization, removal of special characters and stop words, and creation of a small context around the lemma by considering a window of two words before and after the lemma. The texts were then converted to numerical format using CountVectorizer, and labels were encoded using LabelEncoder.
    - **Example of preprocessing:**
      Original text:
      'The meticulously crafted plan, developed by the experienced team, ensured a seamless and successful product launch.'
      Lemma: **"plan"**
      After preprocessing:
      "meticulously crafted plan developed"
    - **Model:** Yarowsky's algorithm was applied for bootstrapping. A seed set of size 5 was created for each lemma, and a Naive Bayes classifier was trained on this seed. Instances with a confidence interval above 90% were added to the seed set. If the seed set remained unchanged, a new seed set was randomly chosen, and the process repeated.
    - **Results:**

Lemma	plan	climate	focus	term	path
% Accuracy	44.55 %	60 %	53.33 %	53.33 %	53.33 %

Table 2. Results from Bootstrapping

- **Gloss BERT:** For the last task, I have used an online pretrained language model GlossBERT, GlossBERT is a state-of-the-art method for Word Sense Disambiguation (WSD), which is the task of identifying the correct meaning or sense of a word in context

- **Preprocessing:** To utilize GlossBERT, different glosses for the lemma were added. This involved **extracting** all possible definitions from WordNet and replacing the lemma with the gloss, resulting in various texts for each lemma.
- **Example of preprocessing:**
  Original text:
  "U.N. group draft plan to reduce emission"
  Gloss – sentiment pair:
  'U.N. group draft plan to reduce emission [SEP] **any number of entities (members) considered as a unit**' → **'group.n.01'**
  'U.N. group draft plan to reduce emission [SEP] (chemistry) **two or more atoms bound together as a single unit and forming part of a molecule**' → **'group.n.02'**
  'U.N. group draft plan to reduce emission [SEP] **a set that is closed, associative, has an identity element and every element has an inverse**' → **'group.n.03'**
- **Model:** The model is specifically tuned on SamCor 3.0 to perform word-sense-disambiguation, The context with it's gloss is passed through the pretrained language model, the output from the model is the probabilities associated with different glosses of the lemma and I chose the sense with the highest probability as a prediction for the sense for my lemma.
- **Results:**

Dataset	**Dev Set**	**Test Set**
Accuracy	52.06 %	44.55 %

Table 3. Results of GlossBERT

- **Conclusion:** In Tables 1, 2, and 3, I present the outcomes of various models applied to the dataset. Regarding Bootstrapping, direct comparison with the baseline is not feasible due to the distinct origin of the Bootstrapping dataset, which is generated by ChatGPT and markedly differs from the provided dataset. When contrasting the results of GlossBERT with the Lesk algorithm, the online pretrained language model outperforms the latter, as it is specifically trained for the word sense disambiguation task. However, BERT's performance may not be optimal, and several factors could contribute to this. Firstly, the model is fine-tuned on SemCor 3.0, which differs from WordNet. Additionally, there are instances where the predicted sense closely resembles the actual sense, leading to the identification of synonyms as predicted senses. In some cases, the model correctly predicts the sense but associates it with multi-word phrases. For instance, in the test set, the lemma "government" yielded the prediction "American_government," resulting in a lower accuracy. Further investigation is warranted to refine predictions and enhance accuracy. Given these considerations, I anticipate that GlossBERT has the potential to achieve an accuracy exceeding 90% on both sets. Notably, the Lesk algorithm consistently performs the least satisfactorily across all cases. A primary factor contributing to its suboptimal performance is its struggle when words exhibit different meanings based on their part of speech. When a word has multiple senses, each corresponding to a different part of speech, Lesk encounters challenges in distinguishing between them.