Report on Interpolated N-gram Model with Expectation-Maximization Optimization

In this study, an interpolated n-gram language model is implemented and evaluated. The primary objectives were to (1) build a statistical n-gram model using Maximum Likelihood Estimation (MLE), (2) extend this model to an interpolated n-gram model with uniform distribution for interpolation weights (λ), and (3) optimize these λ values using the Expectation-Maximization (EM) algorithm to minimize perplexity on a test dataset.

Implementation Summary

1. **Statistical N-gram Model**:
    - **Model Construction**: An n-gram model (`NGramModel`) was built to capture the probability distribution of n-grams in the training data.
    - **Probabilities**: MLE was used to assign probabilities to n-grams.
    - **Data**: The model was trained on a dataset (`wiki.train.tokens`), and its properties were validated by inspecting trigram probabilities.
    - **Result**: The total number of unique trigrams was found to be 1,353,728.
2. **Interpolated N-gram Model**:
    - **Extension**: An interpolated n-gram model (`InterpolatedNGramModel`) was developed, encapsulating n-gram models for n = 1 to 3.
    - **Interpolation Weights**: Initially, uniform weights were assigned to the n-gram models.
    - **Perplexity Calculation**: The perplexity of this model on the test data (`wiki.test.tokens`) was calculated to be 144.545.
3. **Expectation-Maximization Optimization**:
    - **EM Algorithm**: The `EMInterpolatedNGramModel` class was created to optimize the interpolation weights (λ) using the EM algorithm.
    - **Convergence**: The algorithm iteratively adjusted λ until convergence, with a significant reduction in perplexity.
    - **Final λ Values**: The final optimized weights were λ = [0.25851094, 0.56501545, 0.1764736] for unigram, bigram, and trigram models respectively.
    - **EM Perplexity**: After optimization, the perplexity on the test data reduced to 138.381.

Observations

- **Model Complexity**: The unigram component had the least weight, indicating lower reliance on individual word frequencies compared to adjacent word combinations.
- **Bigram Importance**: The bigram component received the highest weight, suggesting that pairs of consecutive words provide significant contextual information.
- **Perplexity Reduction**: The optimization of interpolation weights using the EM algorithm led to a noticeable decrease in perplexity, implying an improvement in the model's predictive performance.
- **Convergence**: The EM algorithm converged after 19 iterations, with diminishing returns in λ adjustments towards the end, indicating a stable solution.