1. **What is the vocabulary?**

   Latent Dirichlet allocation (LDA) reveals topics in a corpus. Batch approach does not scale.
   - Two solutions: parallel and online inference
   - Online: after observing a minibatch of documents, estimate latent variables

   Existing online approaches share some flaw:
   - Immutable vocabulary, drawn from a fixed Dirichlet distribution
   - Cannot capture the appearance of new words

   Fixed vocabulary conceals when:
   - words are invented, e.g., "e-mailsourcing"
   - words cross languages, e.g., "Gangnam" words cross topics, e.g., "vuvuzelas"

   We replace the Dirichlet distribution over topics with an unbounded set of words, drawn from an infinite vocabulary.

2. **What is the vocabulary?**

   - Infinite vocabulary
   - Parameters trained on a English dictionary.

3. **Base Distribution Intuition**

   Base Distribution: Character n-gram Model

   - Parameters trained on a English dictionary.
   - A Dirichlet process provides a distribution over an unbounded set of words (atoms).

4. **Base Distribution Intuition**

   - Parameters trained on a English dictionary.
   - A Dirichlet process provides a distribution over an unbounded set of words (atoms).

5. **Generative Model**

   Generative process of the n-gram character model:
   - Choose a length \( l \in \mathbb{N} \)
   - Iteratively generate a word's \( l \)-th character \( c_l \) given context \( c_{<l} \)

   \[
   G_0(c_1) = \frac{\pi_1}{\sum_{i=1}^{k} \pi_i} \quad G_i(c_{<i}) = \sum_{j=1}^{k} \pi_j G_j(c_{<i-j})
   \]

6. **Variational Distribution**

   Variational distribution is \( q(\theta) = \prod_{n \in \mathbb{N}} \prod_{d \in \mathbb{N}} q(\theta_{d,n}) \).

7. **Truncation Set (TOS)**

   To test the quality of the model, we fit a topic model with 50 topics to the \( G \) from the base distribution.

8. **Generative Process of Online LDA with Infinite Vocabulary**

   - for each topic \( d \)
   - Draw \( \pi_d \) from base distribution \( G_{\text{base}} \)
   - Draw \( \beta_d \) from \( G_{\text{base}} \)
   - Draw \( z_d \) from \( G_{\text{base}} \)
   - Draw \( \theta_d \) from \( G_{\text{base}} \)

9. **Inference Algorithm**

   1. Randomly initialize variational parameters.
   2. Repeat:
      - for each document \( d \) in minibatch do
        - Empirically sample the variational distribution \( q(\theta_d | z_d) \) according to
          \[
          q(\theta_d | z_d) = (1 - \epsilon) \cdot \frac{\lambda_{d,n}}{\sum_{n \in \mathbb{N}} \lambda_{d,n}} + \epsilon \cdot \sum_{n \in \mathbb{N}} \rho_{d,n}
          \]
        - Update variational parameters \( \lambda \) using stochastic gradient descent algorithm
          \[
          \Delta \lambda_{d,n} = 1 - \epsilon \cdot \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{c} \left( \delta(c) \cdot \delta(n) \cdot \delta(d) \right) - \epsilon \cdot \lambda_{d,n}
          \]
          - Update the ranking score according to
            \[
            R_d(\theta_d | z_d) = 1 + \epsilon \cdot \sum_{n \in \mathbb{N}} \rho_{d,n} - R_d(\theta_d | z_d)
            \]
          - Contract vocabulary for every topic if necessary
          - until model convergence

10. **Results: Invariant New Words**

    - Microsoft releases new Xbox console ...
    - The stock reached ...
    - New words are added to the TOS as they appear.
    - After observing \( U \) minibatches, we use a heuristic inspired by Chinese restaurant process to reorder the words in the TOS according to
      \[
      R_d(\theta_d | z_d) = 1 + \epsilon \cdot \sum_{n \in \mathbb{N}} \rho_{d,n} - R_d(\theta_d | z_d)
      \]
    - Retain only the \( T \) terms (truncation size) according to the ranking score.

    - Reordering level \( T \) terms (truncation size) according to the ranking score.

    - Our previous information (e.g., rank and variational parameters) is discarded.

**References**