Assignment 1

Tanishka Nama

21084027

Electrical Part 4 IDD

The objective of this assignment was to explore Zipf's Law in two languages—English and Hindi—using the provided datasets. Alongside this analysis, stemming was performed on the text data, using the Porter Stemmer for English and the Snowball Stemmer for Hindi.

The dataset for English consisted of articles from *The Telegraph* newspaper. The text was processed to analyze word frequency and rank, and the relationship between them was visualized on a log-log graph. Stemming was applied using the Porter Stemmer to group words with the same root.

For Hindi, the dataset included text from *Amar Ujala* and *Jagran* newspapers. The same processing steps were followed, and a log-log graph was created to show how word frequency and rank relate. Stemming for Hindi was performed using the Snowball Stemmer.

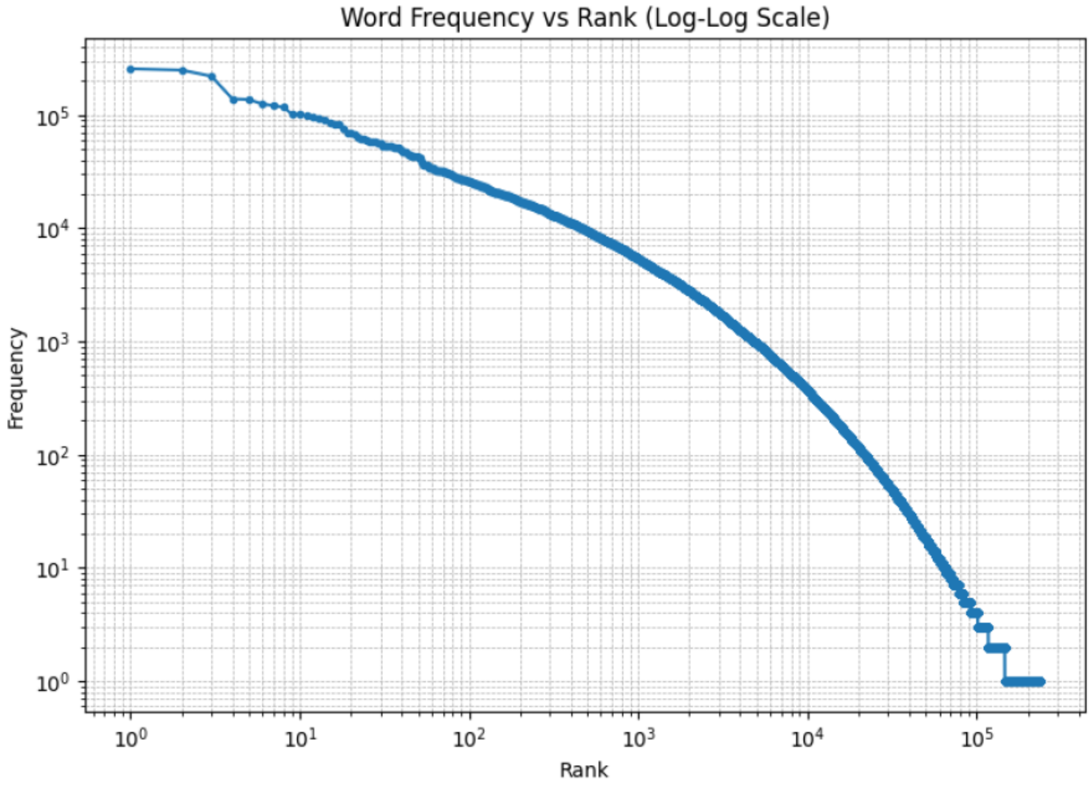
Zipf's Law Analysis:
The graphs for both English and Hindi show patterns consistent with Zipf's Law, which describes the inverse relationship between word rank and frequency. This is captured in the formula:

$f(r) = c / (r^s)$

Where:

- $f(r)$ is the frequency of a word at rank r
- r is the word's rank
- c is a constant
- s is the Zipf exponent

To illustrate this, a reference line representing Zipf's Law ($f(r) = 1/r$) was added for comparison. The graphs were plotted on a log-log scale using Matplotlib's loglog function, and the results visually confirm that the data aligns with Zipf's Law.



Stemming:

English:

Original Unique Words: 234964

Stemmed Unique Words: 179366

Top 5 original words:

said: 256976

pm: 250204

calcutta: 221258

telegraph: 139215

cal: 137542

Top 5 stemmed words:

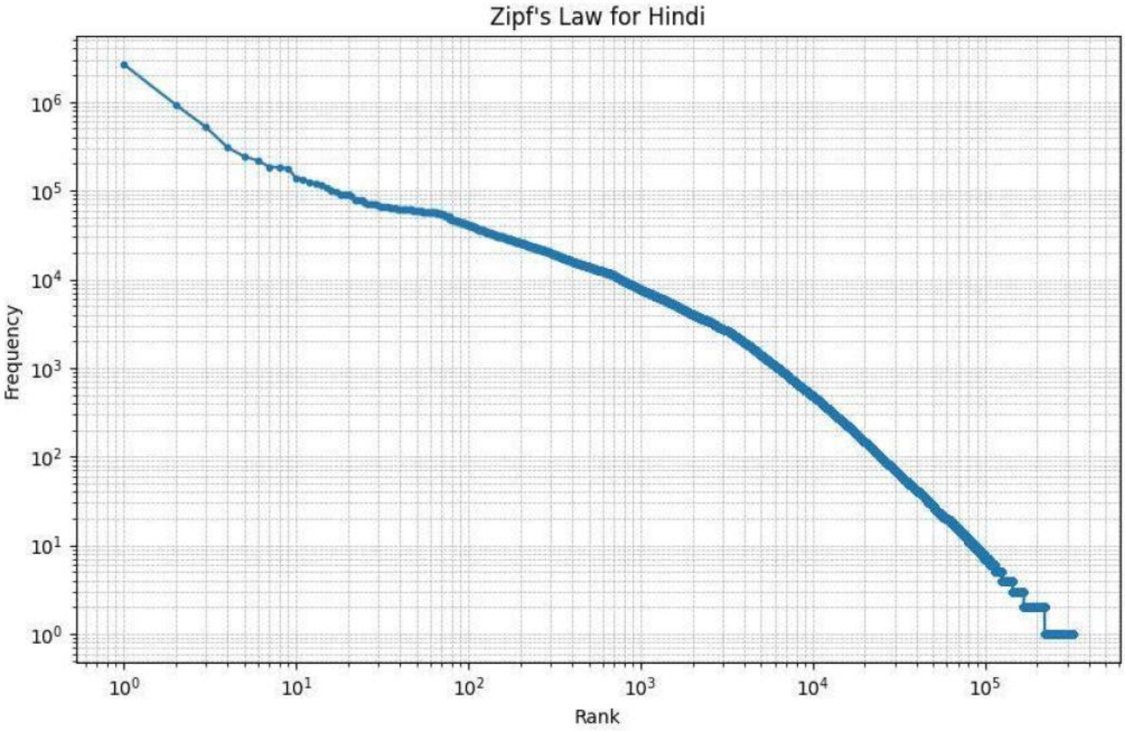
said: 256978

pm: 250310

calcutta: 221916

telegraph: 139444

cal: 137543



Hindi:

Original Unique Words: 321592

Stemmed Unique Words: 269590

Top 5 original words:

के: 2659180

है।: 931999

जागरण्ा: 520872

अाैर: 307291

नहीं: 242416

Top 5 stemmed words:

क: 2667034

है।: 932004

जागरण्: 520872

नह: 461492

अाैर: 307527

Stemming Rules with Examples

Stemming is reducing words to their root form by removing suffixes while keeping the core meaning intact. Here’s how stemming works for both English and Hindi using the **Porter Stemmer** and **Snowball Stemmer**, respectively.

English Stemming Rules (Porter Stemmer)

The **Porter Stemmer** follows a series of steps to remove common suffixes in English while preserving the root word. Below are some key rules with examples:

1. Removing Plural Forms and -ed/-ing Suffixes

- **Rule:**
  - Remove "s" if it appears at the end of a word (except "ss").
  - Remove "ed" or "ing" if the root word remains valid.
- **Examples:**

    - **cats → cat**
    - **ponies → poni**
    - **caressed → caress**
    - **singing → sing**

2. Changing "y" to "i" When Preceded by a Consonant

- **Rule:**
    - If a word ends in "y" and is preceded by a consonant, replace "y" with "i".
- **Examples:**
    - **crying → cri**
    - **happily → happili**

3. Handling Double Suffixes

- **Rule:**
    - Reduce complex suffixes like "-ational" and "-ization" to simpler forms.
- **Examples:**
    - **relational → relate**
    - **conditional → condition**
    - **rational → ration**

4. Removing Suffixes like -al, -ance, -ence, etc.

- **Rule:**
    - Remove common suffixes such as "-al", "-ance", "-ence", "-er".
- **Examples:**
    - **acceptance → accept**
    - **difference → differ**

5. Removing the Final "e"

- **Rule:**
    - If a word ends with a silent "e", remove it.

- **Examples:**
    - **probate → probat**
    - **rate → rat**

Example Workflow:

1. Input: **"Running"**
2. Step 1: Remove "ing" → **"run"**
3. Output: **"run"**

Hindi Stemming Rules (Snowball Stemmer)

The **Snowball Stemmer for Hindi** reduces words to their base forms by removing grammatical suffixes. Here are some key rules with examples:

1. Removing Gender-Based Suffixes

- **Rule:**
    - Remove gender-related suffixes like "ीी" (-ii), "ेे" (-e), "ोो" (-o), "ुु" (-u).
- **Examples:**
    - **छोटी (chhoṭi) → छोट (chhoṭ)**
    - **पसंदे (pasande) → पसंद (pasand)**

2. Removing Plural Suffixes

- **Rule:**
    - Remove plural markers such as "ों" (-on) and "ओं" (-on).
- **Examples:**
    - **किताबों (kitaabon) → किताब (kitaab)**
    - **लोगों (logon) → लोग (log)**

3. Removing Case Suffixes

- **Rule:**

  - Remove oblique case suffixes like "ों" (-on), "ो" (-o), "े" (-e).
- **Examples:**
  - **पाठों (paathon) → पाठ (paath)**
  - **ग्रामों (graamon) → ग्राम (graam)**

4. Removing Verb Suffixes

- **Rule:**
  - Remove verb endings like "ने" (-ne), "ना" (-na), "ती" (-ti), "ते" (-te).
- **Examples:**
  - **खेलने (khelne) → खेल (khel)**
  - **खेली (kheli) → खेल (khel)**

Example Workflow:

1. Input: **"खेलने (khelne)"**
2. Step 1: Remove "ने" suffix → **"खेल (khel)"**
3. Output: **"खेल (khel)"**