CS5760 – Natural Language Processing
Homework 1

D. Sri Kathyayini - 700769996

Q1. Regex – Answers

1 & 2. U.S. ZIP Codes

```
\b\d{5}(?:[- ]\d{4})?\b
```

3. Words that do NOT start with a capital letter

```
\b(?![A-Z])[a-zA-Z]+(?:['-][a-zA-Z]+)*\b
```

4. Numbers (optional sign, commas, decimals, scientific notation)

```
[+-]?(?:\d{1,3}(?:,\d{3})*|\d+)(?:\.\d+)?(?:[eE][+-]?\d+)?
```

5. Email spelling variants

```
(?i)\be[\s\--]?mail\b
```

6. Interjection go / goo / gooo with punctuation

```
\bgo+\b[!.,?]?
```

7. Lines ending with a question mark

```
\?\s*[\)\"’'\]]*\s*$
```

Q2. Byte Pair Encoding (BPE)

2.1 Manual BPE on a Toy Corpus

Corpus:
low low low low low lowest lowest newer newer newer newer newer newer wider wider wider new new

Corpus with end-of-word marker _:
l o w _
l o w _
l o w _
l o w _

l o w _
l o w e s t _
l o w e s t _
n e w e r _
n e w e r _
n e w e r _
n e w e r _
n e w e r _
n e w e r _
w i d e r _
w i d e r _
w i d e r _
n e w _
n e w _

Initial Vocabulary:
{ l, o, w, e, s, t, n, r, i, d, _ }

Step 1:
Most frequent pair: l o
Merge: l o → lo
New token: lo
Updated vocabulary: { lo, w, e, s, t, n, r, i, d, _ }

Step 2:
Most frequent pair: lo w
Merge: lo w → low
New token: low
Updated vocabulary: { low, e, s, t, n, r, i, d, _ }

Step 3:
Most frequent pair: low _
Merge: low _ → low_
New token: low_
Updated vocabulary: { low_, e, s, t, n, r, i, d, _ }

2.2 Mini-BPE Learner (Conceptual Output)

Top pairs learned sequentially:
1. l o
2. lo w
3. low _

Vocabulary size increases after each merge.

Segmentation examples:
new → n e w _
newer → n e w er_
lowest → low est_
widest → wid est_
newestest → new est est_

Explanation:
Subword tokenization helps solve the OOV problem by decomposing unseen words into known smaller units.
Even if a word like "newestest" was never seen, it can be segmented using learned subwords.
Subwords often correspond to meaningful morphemes such as suffixes.
For example, the token "er_" aligns with the English comparative/agent suffix.
This allows models to generalize better to unseen forms.

2.3 BPE on a Paragraph (English)

Paragraph:
Natural language processing enables machines to understand human language.
It involves syntax, semantics, and learning patterns from data.
Modern systems rely heavily on subword tokenization techniques.
These techniques improve robustness and reduce vocabulary size.

Frequent merges:
1. t h
2. th e
3. ing_
4. sub word
5. token ization

Longest subword tokens:
processing_
tokenization_
techniques_
understand_
language_

Segmentation examples:
processing → process ing_
techniques → tech niques_
tokenization → token ization_
robustness → robust ness_
semantics → semantic s_

Reflection:
BPE learns prefixes, suffixes, stems, and sometimes whole words.
Suffixes like ing_ and s_ are commonly learned.
A major advantage is handling rare and unseen words effectively.
Another benefit is reduced vocabulary size.
However, subwords may break semantic meaning in some cases.
Additionally, token boundaries may not always align with true linguistic units.

Q3. Bayes' Rule Applied to Text Classification

The classifier selects the class c that maximizes:
$c_{MAP} = \arg\max P(c)\, P(d \mid c)$

1. Explanation of Terms

P(c) — Prior Probability of the Class:
This represents the probability of a class before observing the document.
It is estimated from training data as the proportion of documents in class c.
It captures how common a class is overall.

P(d | c) — Likelihood of the Document Given the Class:
This is the probability of observing the document assuming it belongs to class c.
In text classification, it is computed using word probabilities learned from that class.
It measures how well the document matches the language patterns of class c.

P(c | d) — Posterior Probability of the Class Given the Document:
This is the probability that the document belongs to class c after observing it.
It combines the prior probability with evidence from the document.
The goal of classification is to select the class with the highest posterior probability.

2. Why the Denominator P(d) Can Be Ignored

Bayes' Rule states:
$P(c \mid d) = (P(d \mid c)\, P(c)) / P(d)$

The denominator P(d) represents the probability of the document.
It is the same for all classes and does not depend on c.
Since P(d) is constant, it does not affect class comparison.
Therefore, we can ignore P(d) and compare only $P(c)\, P(d \mid c)$.

Q4. Add-1 (Laplace) Smoothing

Given:
P(-) = 3/5, P(+) = 2/5
Vocabulary size |V| = 20
Total token count in negative class N- = 14

1. Denominator for Likelihood Estimation

Using add-1 smoothing, the denominator is:
N- + |V|
= 14 + 20
= 34

2. Likelihood Computations

(a) P(predictable | -)
Count of 'predictable' in negative documents = 2

P(predictable | -) = (2 + 1) / (14 + 20)
= 3 / 34

(b) P(fun | -)
Count of 'fun' in negative documents = 0

P(fun | -) = (0 + 1) / (14 + 20)
= 1 / 34

Final Answers Summary

Denominator = 34
P(predictable | -) = 3/34

P(fun | -) = 1/34