Noah Buchanan
Problem Set 1
Information Retrieval at 5:25 PM

June 23, 2021

Summary output

```
======= Project EWOK Tokenization Summary =======

Total Number of Tokens:                    712896246
Files Processed:                           839918
Emails Found:                              448018
Prices Found:                              1249069
Domains Found:                             52437340
Words Found:                               386243010
Phone Numbers Found:                       9471882


real     1548m5
230s
user     1414m52
325s
sys      14m9
181s
```

Source Code: PS1Tokenizer.jj

```
options
{
}

/* Generate UAFSTokenizer */
PARSER_BEGIN(UATokenizer)
import java.io.*;
```

```
public class UATokenizer
{
  public static void main(String [] args) throws ParseException
  {
    int domaincount = 0;
    int emailcount = 0;
    int phonecount = 0;
    int pricecount = 0;
    int wordcount = 0;
    int tokencount = 0;
    int filecount = 0;
    int subfoldercount = 0;
    try
    {
      File f = new File(args [0]);
      File [] FileList = f.listFiles();
      BufferedInputStream bis = new BufferedInputStream(new FileInputStream(File
      UATokenizer parser = new UATokenizer(bis);
      BufferedOutputStream tokens = new BufferedOutputStream(new FileOutputStream
      BufferedOutputStream emails = new BufferedOutputStream(new FileOutputStream
      BufferedOutputStream phones = new BufferedOutputStream(new FileOutputStream
      BufferedOutputStream domains = new BufferedOutputStream(new FileOutputStrea
      BufferedOutputStream prices = new BufferedOutputStream(new FileOutputStream
      BufferedOutputStream words = new BufferedOutputStream(new FileOutputStream
      File subfolder = null;
      for (File file : FileList)
      {
        File [] subfiles = file.listFiles();
        for (File subfile : subfiles)
        {
          if (filecount % 1500 == 0)
          {
            subfolder = new File(args [1] + "/" + "sub" + String.format("%04d",s
            subfolder.mkdir();
            subfoldercount++;
          }
          filecount++;
          bis = new BufferedInputStream(new FileInputStream(subfile.getPath()));
          BufferedOutputStream bos1 = new BufferedOutputStream(new FileOutputStre
          parser.ReInit(bis);
          Token t = getNextToken();
          while (t.kind != UATokenizer.EOF)
          {
            tokencount++;
            byte [] b = String.format("Type : %-10s Token: %s %n", UATokenizer.t
            bos1.write(b);
            tokens.write(b);
            b = String.format(”Token: %s %n”, t.image).getBytes();
            if (UATokenizer.tokenImage [t.kind].toString().equals(”<WORD>”))
            {
              words.write(b);
              wordcount++;
            }
            else if (UATokenizer.tokenImage [t.kind].toString().equals(”<EMAIL>”
            {
              emails.write(b);
              emailcount++;
            }
            else if (UATokenizer.tokenImage [t.kind].toString().equals(”<PHONENU
            {
              phones.write(b);
              phonecount++;
            }
            else if (UATokenizer.tokenImage [t.kind].toString().equals(”<DOMAIN>
            {
              domains.write(b);
              domaincount++;
            }
            else if (UATokenizer.tokenImage [t.kind].toString().equals(”<PRICE>”
            {
              prices.write(b);
              pricecount++;
            }
            t = parser.getNextToken();
          }
          bos1.close();
          bis.close();
        }
      }
      tokens.close();
      words.close();
      domains.close();
      prices.close();
      phones.close();
      emails.close();
      BufferedOutputStream summary = new BufferedOutputStream(new FileOutputStrea
      summary.write((”======= Project EWOK Tokenization Summary =======\n\n”).ge
      summary.write((”Total Number of Tokens:
” + tokencount + ”\n”).getBytes());
      summary.write((”Files Processed:
” + filecount + ”\n”).getBytes());
```

```
      summary.write((”Emails Found:
” + emailcount + ”\n”).getBytes());
      summary.write((”Prices Found:
” + pricecount + ”\n”).getBytes());
      summary.write((”Domains Found:
” + domaincount + ”\n”).getBytes());
      summary.write((”Words Found:
” + wordcount + ”\n”).getBytes());
      summary.write((”Phone Numbers Found:
” + phonecount + ”\n\n”).getBytes());
      summary.close();
    }
    catch (Exception ex)
    {
      ex.printStackTrace();
    }
    ;
  }
}

PARSER_END(UATokenizer)

TOKEN_MGR_DECLS :
{
  void CommonTokenAction(Token t)
  {
  /* System.out.println(”Token : ” + t.image); */
  }
}

SKIP :
{
  ” ”
| ”\n”
| ”\t”
| ”\r”
| < USELESSINFOTAGS :
    (
      (
        ”<style” (~[ ])* ”style>”
      )
    |
      (
        ”<script” (~[ ])* ”script>”
      )
    ) >
```

```
| < TAG :
    (
      "<" ([ "A"-"Z", "a"-"z", "0"-"9", "\"", "'", "=", ":", "/", "?", "-", ",",
    ) >
}

/* JavaCC syntax */
TOKEN :
{
  < EMAIL :
    ([ "a"-"z", "A"-"Z", "0"-"9", ".", "_", "-", "!", "#", "$", "%", "&", "'", "
    "^", "=", "{", "|", "}", "~" ])+ "@"
    (
      ([ "a"-"z", "A"-"Z" ])+ "."
    )+
    ([ "a"-"z", "A"-"Z" ])+ >
| < DOMAIN :
    ([ "a"-"z", "A"-"Z" ])+ "."
    (
      ([ "a"-"z", "A"-"Z", "0"-"9" ])+ "."
    )+
    ([ "a"-"z", "A"-"Z" ])+ >
| < PRICE :
    "$" ([ "0"-"9" ])
    {
      1, 3
    }
    (
      (",")? ([ "0"-"9" ])
      {
        3
      }
    )*
    (
      "." ([ "0"-"9" ])
      {
        1, 2
      }
    )? >
| < PHONENUMBER :
    (
      ("+")?
      ([ "0"-"9" ])
      {
        1, 3
      }
```

```
        ([ ” ”, ”−”, ”.”, ”\r” ])
     )?
     (
        (
           (”(”) ([ ”0”−”9” ])
           {
              3
           }
           (”)”)
        )
     |  ([ ”0”−”9” ])
        {
           3
        }
     )
     ([ ” ”, ”−”, ”.”, ”\r” ]) ([ ”0”−”9” ])
     {
        3
     }
     ([ ” ”, ”−”, ”.”, ”\r” ]) ([ ”0”−”9” ])
     {
        4
     }
  >
| < WORD : ([ ”a”−”z”, ”A”−”Z” ])
     {
        1, 20
     }
  >
| < NUMBER :
     ([ ”0”−”9” ])
     {
        1, 3
     }
     (
        (”,”)? ([ ”0”−”9” ])
        {
           3
        }
     )*
     (
        ”.” ([ ”0”−”9” ])
        {
           1,
        }
     )? >
```

```
| < OTHER : ~[ ] >
}
```