In [6]:

```
import os
import json
from pathlib import Path
import zipfile
import email
from email.policy import default
from email.parser import Parser
from datetime import timezone
from collections import namedtuple

import pandas as pd
import s3fs
from bs4 import BeautifulSoup
from dateutil.parser import parse
from chardet.universaldetector import UniversalDetector

from pyspark.ml import Pipeline
from pyspark.ml.feature import CountVectorizer
from pyspark.ml.feature import HashingTF, Tokenizer
from pyspark.sql import SparkSession
from pyspark.sql.functions import col
from pyspark.ml.pipeline import Transformer
from pyspark.sql.functions import udf
from pyspark.sql.types import StructType, StringType, StructField

import pandas as pd

current_dir = Path(os.getcwd()).absolute()
results_dir = current_dir.joinpath('results')
results_dir.mkdir(parents=True, exist_ok=True)
data_dir = current_dir.joinpath('data')
data_dir.mkdir(parents=True, exist_ok=True)
enron_data_dir = data_dir.joinpath('enron')

output_columns = [
        'payload',
        'text',
        'Message_D',
        'Date',
        'From',
        'To',
        'Subject',
        'Mime-Version',
        'Content-Type',
        'Content-Transfer-Encoding',
        'X-From',
        'X-To',
        'X-cc',
        'X-bcc',
        'X-Folder',
        'X-Origin',
        'X-FileName',
        'Cc',
        'Bcc'
]
```

```
columns = [column.replace('-', '_') for column in output_columns]

ParsedEmail = namedtuple('ParsedEmail', columns)

spark = SparkSession\
    .builder\
    .appName("Assignment04")\
    .getOrCreate()
```

In [ ]:

Copied enron file. Hence, below is commented

In [3]:
```
#def copy_data_to_local():
#    dst_data_path = data_dir.joinpath('enron.zip')
#    endpoint_url='https://storage.budsc.midwest-datascience.com'
#    enron_data_path = 'data/external/enron.zip'
#
#    s3 = s3fs.S3FileSystem(
#        anon=True,
#        client_kwargs={
#            'endpoint_url': endpoint_url
#        }
#    )
#
#
#    s3.get(enron_data_path, str(dst_data_path))
#
#    with zipfile.ZipFile(dst_data_path) as f_zip:
#        f_zip.extractall(path=data_dir)
#
#copy_data_to_local()
```

Assignment 4.1

In [7]:
```
def read_raw_email(email_path):
    detector = UniversalDetector()

    try:
        with open(email_path) as f:
            original_msg = f.read()
    except UnicodeDecodeError:
        detector.reset()
        with open(email_path, 'rb') as f:
            for line in f.readlines():
                detector.feed(line)
                if detector.done:
                    break
        detector.close()
        encoding = detector.result['encoding']
        with open(email_path, encoding=encoding) as f:
            original_msg = f.read()

    return original_msg

def make_spark_df():
    records = []
    for root, dirs, files in os.walk(enron_data_dir):
        for file_path in files:
            ## Current path is now the file path to the current email.
            ## Use this path to read the following information
            ## original_msg
            ## username (Hint: It is the root folder)
            ## id (The relative path of the email message)
            current_path = Path(root).joinpath(file_path)
            record = {}
            username = os.path.basename(os.path.dirname(root))
            id = username+"/"+os.path.basename(root)+"/"+file_path
            record["id"]=id
            record["username"]=username
            record["original_msg"]=read_raw_email(current_path)
            records.append(record)
    ## TODO: Complete the code to code to create the Spark dataframe
    schemaString = "id username original_msg"
    fields = [StructField(field_name, StringType(), True) for field_name in
    schema = StructType(fields)
    return spark.createDataFrame(records, schema)

df = make_spark_df()
```

In [8]:
```
df.show()
```

```
+--------------------+---------------+--------------------+
|                  id|       username|        original_msg|
+--------------------+---------------+--------------------+
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <1068...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <1807...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <8502...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <3244...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <2159...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <1622...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <2744...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <1334...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <2054...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <7445...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <3152...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <7621...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <1608...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <3203...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <2286...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <2494...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <3258...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <8555...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <2420...|
|mims-thurston-p/d...|mims-thurston-p|Message-ID: <7314...|
+--------------------+---------------+--------------------+
only showing top 20 rows
```

In [9]:
```
df.printSchema()
```

```
root
 |-- id: string (nullable = true)
 |-- username: string (nullable = true)
 |-- original_msg: string (nullable = true)
```

Assignment 4.2

In [10]:
```
plain_msg_example = """
Message-ID: <6742786.1075845426893.JavaMail.evans@thyme>
Date: Thu, 7 Jun 2001 11:05:33 -0700 (PDT)
From: jeffrey.hammad@enron.com
To: andy.zipper@enron.com
Subject: Thanks for the interview
Mime-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
X-From: Hammad, Jeffrey </O=ENRON/OU=NA/CN=RECIPIENTS/CN=NOTESADDR/CN=CBBE3
X-To: Zipper, Andy </O=ENRON/OU=NA/CN=RECIPIENTS/CN=AZIPPER>
X-cc:
X-bcc:
X-Folder: \Zipper, Andy\Zipper, Andy\Inbox
X-Origin: ZIPPER-A
X-FileName: Zipper, Andy.pst

Andy,

Thanks for giving me the opportunity to meet with you about the Analyst/ As

Thanks and Best Regards,

Jeff Hammad
"""

html_msg_example = """
Message-ID: <21013632.1075862392611.JavaMail.evans@thyme>
Date: Mon, 19 Nov 2001 12:15:44 -0800 (PST)
From: insynconline.6jy5ympb.d@insync-palm.com
To: tstaab@enron.com
Subject: Last chance for special offer on Palm OS Upgrade!
Mime-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
X-From: InSync Online <InSyncOnline.6jy5ympb.d@insync-palm.com>
X-To: THERESA STAAB <tstaab@enron.com>
X-cc:
X-bcc:
X-Folder: \TSTAAB (Non-Privileged)\Staab, Theresa\Deleted Items
X-Origin: Staab-T
X-FileName: TSTAAB (Non-Privileged).pst

<html>

<html>
<head>
<title>Paprika</title>
<meta http-equiv="Content-Type" content="text/html;">
</head>
<body bgcolor="#FFFFFF" TEXT="#333333" LINK="#336699" VLINK="#6699cc" ALINK
<table border="0" cellpadding="0" cellspacing="0" width="582">
<tr valign="top">
  <td width="582" colspan="9"><nobr><a href="http://insync-online.p04.com/u
</tr>
<tr valign="top">
```

```
  <td width="4" bgcolor="#CCCCCC"><img src="http://images4.postdirect.com/m
  <td width="20"><img src="http://images4.postdirect.com/master-images/4047
  <td width="165"><br><a href="http://insync-online.p04.com/u.d?LkReaQA5ecz
  <td width="20"><img src="http://images4.postdirect.com/master-images/4047
  <td width="165"><br><a href="http://insync-online.p04.com/u.d?BkReaQA5ecz
  <td width="20"><img src="http://images4.postdirect.com/master-images/4047
  <td width="165"><br><a href="http://insync-online.p04.com/u.d?JkReaQA5ecz
  <td width="19"><img src="http://images4.postdirect.com/master-images/4047
  <td width="4" bgcolor="#CCCCCC"><img src="http://images4.postdirect.com/m
</tr>
</table>
<table border="0" cellpadding="0" cellspacing="0" width="582">
<tr valign="top">
  <td width="4" bgcolor="#CCCCCC"><img src="http://images4.postdirect.com/m
  <td width="574"><br>
    <table border="0" cellpadding="0" cellspacing="0" width="574" bgcolor="
    <tr>
      <td width="50"><img src="http://images4.postdirect.com/master-images/
      <td width="474"><font face="verdana, arial" size="-2"color="#000000">
        <br>
        Dear THERESA,
        <br><br>
        Due to overwhelming demand for the Palm OS&#174; v4.1 Upgrade with
        extending the special offer of 25% off through November 30, 2001. S
        increase the functionality of your Palm&#153; III, IIIx, IIIxe, III
        new Palm OS v4.1 through this extended special offer. You'll receiv
        <b>for just $29.95 when you use Promo Code <font color="#FF0000">OS
        <b>$10 savings</b> off the list price.
        <br><br>
        <a href="http://insync-online.p04.com/u.d?NkReaQA5eczXRh=51">Click
        <br><br>
        <a href="http://insync-online.p04.com/u.d?MkReaQA5eczXRm=61"><img s
        <br><br>
        You can do a lot more with your Palm&#153; handheld when you upgrad
        favorite features just got even better and there are some terrific
        <br><br>
        <LI> Handwrite notes and even draw pictures right on your Palm&#153
        <LI> Tap letters with your stylus and use Graffiti&#174; at the sam
        <LI> Improved Date Book functionality lets you view, snooze or clea
        <LI> You can easily change time-zone settings</LI>

        <br><br>
        <a href="http://insync-online.p04.com/u.d?WkReaQA5eczXRb=71"><img s
        <br><br>
        <LI> <nobr>Mask/unmask</nobr> private records or hide/unhide direct
        <LI> Lock your device automatically at a designated time using the
        <LI> Always remember your password with our new Hint feature*</LI>

        <br><br>
        <a href="http://insync-online.p04.com/u.d?VEReaQA5eczXRQ=81"><img s
        <br><br>
        <LI> Use your GSM compatible mobile phone or modem to get online an
        <LI> Stay connected with email, instant messaging and text messagin
        <LI> Send applications or records through your cell phone to schedu
             important information to others</LI>

        <br><br>
```

```
        All this comes in a new operating system that can be yours for just
        upgrade to the new Palm&#153; OS v4.1</a> and you'll also get the l
        <nobr>1-800-881-7256</nobr> to order via phone.
        <br><br>
        Sincerely,<br>
        The Palm Team
        <br><br>
        P.S. Remember, this extended offer opportunity of 25% savings absol
        and is only available through the Palm Store when you use Promo Cod
        <br><br>
        <img src="http://images4.postdirect.com/master-images/404707/bottom
        <br><img src="http://images4.postdirect.com/master-images/404707/cl
        </font></td>
      <td width="50"><img src="http://images4.postdirect.com/master-images/
    </tr>
    </table></td>
    <td width="4" bgcolor="#CCCCCC"><img src="http://images4.postdirect.com
  </tr>
  <tr>
  <td colspan="3"><img src="http://images4.postdirect.com/master-images/404
  </tr>
</table>
<table border="0" cellpadding="0" cellspacing="0" width="582">
  <tr>
    <td width="54"><img src="http://images4.postdirect.com/master-images/40
    <td width="474"><font face="arial, verdana" size="-2" color="#000000"><
    * This feature is available on the Palm&#153; IIIx, Palm&#153; IIIxe, a
    ** Note: To use the MIK functionality, you need either a Palm OS&#174;
    with  <nobr>built-in</nobr> modem or data capability that has either an
    are using a phone, you must have data services from your mobile service
    a list of tested and supported phones that you can use with the MIK. Ca
    <br><br>
    ------------------<br>
    To modify your profile or unsubscribe from Palm newsletters, <a href="h
    Or, unsubscribe by replying to this message, with "unsubscribe" as the
    <br><br>
    ------------------<br>
    Copyright&#169; 2001 Palm, Inc. Palm OS, Palm Computing, HandFAX, HandS
    HotSync, iMessenger, MultiMail, Palm.Net, PalmConnect, PalmGlove, PalmM
    and the Palm Platform Compatible Logo are registered trademarks of Palm
    AnyDay, EventClub, HandMAIL, the HotSync Logo, PalmGear, PalmGlove, Pal
    trade dress, PalmSource, Smartcode, and Simply Palm are trademarks of P
    product names may be trademarks or registered trademarks of their respe
    <img src="http://images4.postdirect.com/master-images/404707/clear.gif"
    <td width="54"><img src="http://images4.postdirect.com/master-images/40
  </tr>
</table><br><br><br><br>
<!-- The following image is included for message detection -->
<img src="http://p04.com/1x1.dyn" border="0" alt="" width="1" height="1">
<img src="http://p04.com/1x1.dyn?0vEGou8Ig30ba2L2bLn" width=1 height=1></bo
</html>

</html>
"""
plain_msg_example = plain_msg_example.strip()
html_msg_example = html_msg_example.strip()
```

In [12]:
```
def parse_html_payload(payload):
    """
    This function uses Beautiful Soup to read HTML data
    and return the text.  If the payload is plain text, then
    Beautiful Soup will return the original content
    """
    soup = BeautifulSoup(payload, 'html.parser')
    return str(soup.get_text()).encode('utf-8').decode('utf-8')

def parse_email(original_msg):
    result = {}
    msg = Parser(policy=default).parsestr(original_msg)
    ## TODO: Use Python's email library to read the payload and the headers
    ## https://docs.python.org/3/library/email.examples.html
    result['payload'] = msg.get_payload()
    result['text'] = parse_html_payload(result['payload'])
    try:
        for key, value in msg.items():
            result[key.replace('-', '_')] = value
    except Exception as e:
        print('Problem parsing email: {}\n{}'.format(email_path, e))
    try:
        result['Date'] = parse(result['Date'], ignoretz=False).isoformat()
    except Exception as e:
        print('Problem converting date: {}\n{}'.format(result.get('date'),
    tuple_result = tuple([str(result.get(column, None)) for column in colum
    return ParsedEmail(*tuple_result)
```

In [13]:
```
parsed_msg = parse_email(plain_msg_example)
```

In [14]:
```
print(parsed_msg.text)
```

```
Andy,

Thanks for giving me the opportunity to meet with you about the Analyst/
Associate program.  I enjoyed talking to you, and look forward to contrib
uting to the success that the program has enjoyed.

Thanks and Best Regards,

Jeff Hammad
```

In [15]:
```
parsed_html_msg = parse_email(html_msg_example)
```

In [16]:
```
print(parsed_html_msg.text)
```

```
    * This feature is available on the Palm™ IIIx, Palm™ IIIxe, and Pal
m™ Vx.
    ** Note: To use the MIK functionality, you need either a Palm OS® c

ompatible modem or a phone
    with  built-in modem or data capability that has either an infrared
port or cable exits.  If you
```

Assignment 4.3

In [17]:
```
## This creates a schema for the email data
email_struct = StructType()

for column in columns:
    email_struct.add(column, StringType(), True)
```

In [18]:

```
## This creates a user-defined function which can be used in Spark
parse_email_func = udf(lambda z: parse_email(z), email_struct)

def parse_emails(input_df):
    new_df = input_df.select(
        'username', 'id', 'original_msg', parse_email_func('original_msg').
    )
    for column in columns:
        new_df = new_df.withColumn(column, new_df.parsed_email[column])

    new_df = new_df.drop('parsed_email')
    return new_df

class ParseEmailsTransformer(Transformer):
    def _transform(self, dataset):
        """
        Transforms the input dataset.

        :param dataset: input dataset, which is an instance of :py:class:`p
        :returns: transformed dataset
        """
        return dataset.transform(parse_emails)

## Use the custom ParseEmailsTransformer, Tokenizer, and CountVectorizer
## to create a spark pipeline
## TODO: Complete code
parseemailtransformer = ParseEmailsTransformer()
tokenizer = Tokenizer(inputCol="text", outputCol="words")
cv = CountVectorizer(inputCol=tokenizer.getOutputCol(), outputCol="features
email_pipeline = Pipeline(stages=[parseemailtransformer, tokenizer, cv])

model = email_pipeline.fit(df)
result = model.transform(df)
```

In [19]:
```
result.select('id', 'words', 'features').show()
```

```
+--------------------+--------------------+--------------------+
|                  id|               words|            features|
+--------------------+--------------------+--------------------+
|mims-thurston-p/d...|[, , >, -----orig...|(108640,[0,4,15,3...|
|mims-thurston-p/d...|[[image], see, me...|(108640,[1,2,3,6,...|
|mims-thurston-p/d...|[continental.com,...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[, , , , , [image...|(108640,[0,6,8,20...|
|mims-thurston-p/d...|[, , , -----origi...|(108640,[0,1,2,10...|
|mims-thurston-p/d...|[, , , -----origi...|(108640,[0,1,9,10...|
|mims-thurston-p/d...|[, david, grant, ...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[, our, natural, ...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[, , , what's, ne...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[, [image], , [im...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[[image], [image]...|(108640,[0,1,3,7,...|
|mims-thurston-p/d...|[continental.com,...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[yes,, i, will, g...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[, , got, a, ques...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[, , , , , , , , ...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[patrice:, could,...|(108640,[0,1,3,6,...|
|mims-thurston-p/d...|[, you, are, rece...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[patrice,, , kazt...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[vonda,=20, , the...|(108640,[0,1,2,3,...|
|mims-thurston-p/d...|[, [image], [imag...|(108640,[0,1,2,3,...|
+--------------------+--------------------+--------------------+
only showing top 20 rows
```

In [ ]: