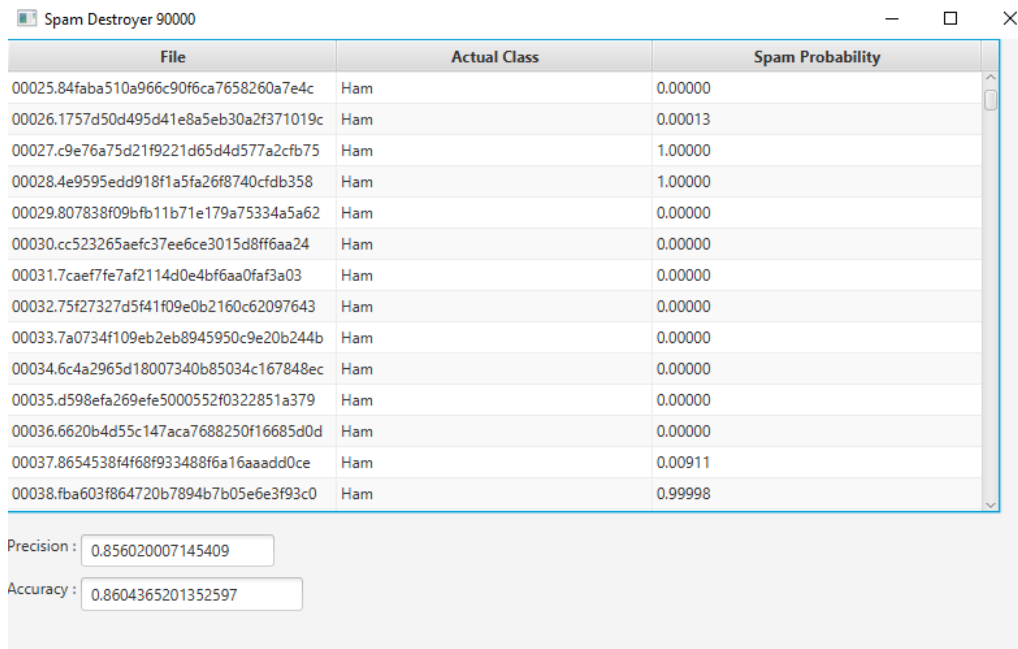


Project Information:

This project utilizes TreeMaps in java to detect if an email is spam or not. The program reads files from a selected directory and analyzes each word in each file as part of the training phase. The program maps each unique word, paired with how many files contain that word into two maps, one for ham files and one for spam files. With these maps the program uses probability laws to calculate the probability if a file is spam, or ham, given a word, for each word. From this calculation, the probability of the file being spam given the file is calculated, done so by logarithmic and exponential equations to normalize the data. The process is tested with test data, which is read through, and spam probability is calculated. Output is a table containing all the files tested, as well as the true class and the spam probability. Accuracy and precision of the calculations were included as well.



The screenshot shows a window titled "Spam Destroyer 90000". It contains a table with three columns: "File", "Actual Class", and "Spam Probability". The table lists 15 files, all of which are classified as "Ham". The spam probabilities range from 0.00000 to 0.99998. Below the table, there are two input fields: "Precision : 0.856020007145409" and "Accuracy : 0.8604365201352597".

File	Actual Class	Spam Probability
00025.84fab510a966c90f6ca7658260a7e4c	Ham	0.00000
00026.1757d50d495d41e8a5eb30a2f371019c	Ham	0.00013
00027.c9e76a75d21f9221d65d4d577a2cfb75	Ham	1.00000
00028.4e9595edd918f1a5fa26f8740cfdb358	Ham	1.00000
00029.807838f09bfb11b71e179a75334a5a62	Ham	0.00000
00030.cc523265aefc37ee6ce3015d8ff6aa24	Ham	0.00000
00031.7caef7fe7af2114d0e4bf6aa0faf3a03	Ham	0.00000
00032.75f27327d5f41f09e0b2160c62097643	Ham	0.00000
00033.7a0734f109eb2eb8945950c9e20b244b	Ham	0.00000
00034.6c4a2965d18007340b85034c167848ec	Ham	0.00000
00035.d598efa269efe5000552f0322851a379	Ham	0.00000
00036.6620b4d55c147aca7688250f16685d0d	Ham	0.00000
00037.8654538f4f68f933488f6a16aaadd0ce	Ham	0.00911
00038.fba603f864720b7894b7b05e6e3f93c0	Ham	0.99998

Precision : 0.856020007145409

Accuracy : 0.8604365201352597

Improvements:

Improvements were made to this process by creating a list of words to be blacklisted from the calculations. Several of the top 10 most common english words were blacklisted, as well as some words that are extremely common in the email domain (ie. 'sent', 'received').

A safety was implemented in the ProbabilityCounter class to prevent illegal math errors in the equations caused by $\ln(0)$ or $\ln(1-1)$. This was prevented by altering the values to be very close to 1 or 0, but not exactly 1 or 0. Doing so caused an increase in accuracy and precision.

The model was modified to reject words that only occurred in a single file from the probability calculation, to remove obscure words or words that act as outliers in a spam case. Doing so increased the accuracy.

How-to-Run:

Clone from https://github.com/Jake-Andrews/csci2020u_jake-andrews.git, and navigate to the csci2020uAssignment1 folder.

Open the project in IntelliJ and navigate to Main.java, from here press run and the program will ask you to select a directory.

Select 'Data' from csci2020uAssignment1/src, then the program will run and output a small window

Important: This program is run on JDK 12.0.1 and javafx 15.0.1

References/Resources/Links:

<https://www.espressoenglish.net/the-100-most-common-words-in-english/>