Experimental Report: Naïve-Bayes Spam Filter
CIS 521: Artificial Intelligence
Zhengxuan Wu

General Design Of The System

This is a spam filter model based strictly on Naïve-Bayes assumption, in which every feature is independent of each other given its class. My system involves: preprocessed tokenized method, unigram feature detection, bigram feature detection, capitalization feature detection, length feature detection and re-enforced rarity feature of word. The initialization time is 13.06 second in local, and the running time for all developed set is 9.97 second in local. The error rate is 0.25% on developed set, and 0.25% on training set.

Pre-processing And Tokenization

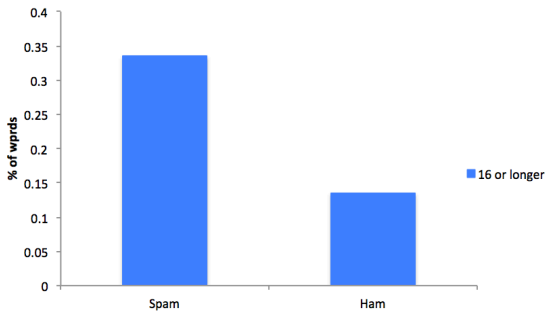
The first round tokenization toke place in separating the email with white spaces and punctuations. The second round, all the punctuation tokens were deleted for reducing the noise. The third round, all the repeated tokens were deleted, so that no same token will be presented in a same email, and this will help the system to comparing emails across the system rather than distracting by a single email. The final round is for deleting all the *stop words (this includes all the email structural words, like 'FONT','SIZE', and special capitalized words, like country names, and common words sharing across spam and ham which will not give us further information but noises, like 'the', 'a', 'and')*.

Experimented Features

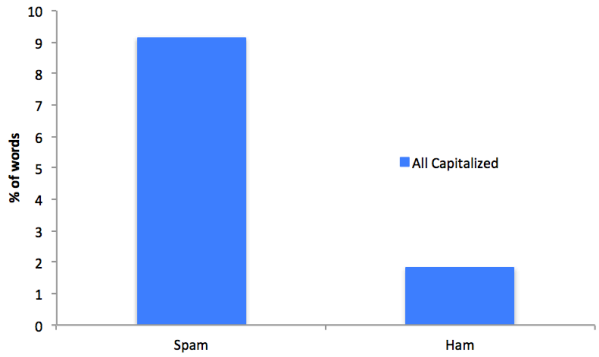
- Unigram and Bigram features
- Length of the words features (Separating length to different bins)
- Capitalization features (All capitalized, 1 capitalized, None capitalized)
- Whether a token is only consisted of number
- Sharing rate of word (Ham tends to have more sharing words)

Discussion Of Features

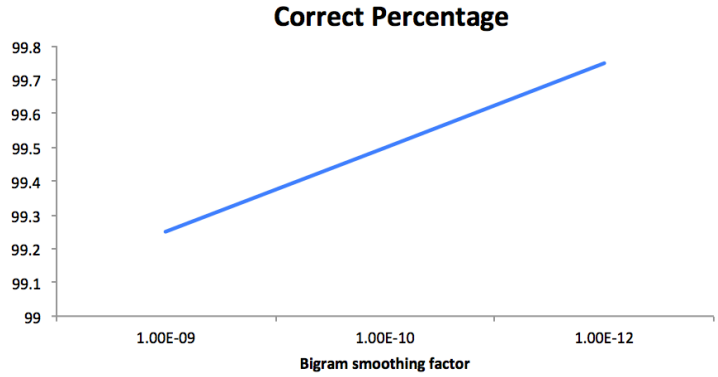
- Length of words
  The length of words including 16 characters or longer differs a lot, as shown.



- Capitalization of words
  All capital case differs the most, and followed by 1 capital character.

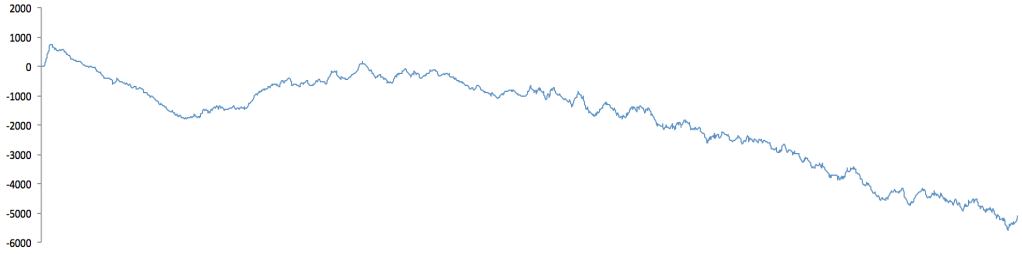


- Tuning smoothing factors
  Different smoothing factors for unigram and bigram and other features. Apparently, the smoothing factor for bigram must be much lower than the smoothing factor for unigram, because we are having a much larger poll for bigram. If the smoothing factor for bigram is too high, then we are adding a very big noise to our system which is bad, as shown.



Discussion Of Error – DEV 283

- The difference between spam log probability and ham log probability



  We can see that only the beginning of the email indicates this is a spam, but a large portion of this email is well-written.
- We can see from the token list of this email, and find that this spam contains a large portion of very well-written words. And this will distract the naïve-Bayes classifier. We can see that only the beginning portion and the ending portion have indicators of spam. This email used a large portion of words to disguise spam characteristics.