require\_once('Category.php');

class NaiveBayesClassifier {

public function \_\_construct() {

}

/\*\*

\* sentence is text(document) which will be classified as ham or spam

\* @return category- ham/spam

\*/

public function classify($sentence) {

// extracting keywords from input text/sentence

$keywordsArray = $this -> tokenize($sentence);

// classifying the category

$category = $this -> decide($keywordsArray);

return $category;

}

/\*\*

\* @sentence- text/document provided by user as training data

\* @category- category of sentence

\* this function will save sentence aka text/document in trainingSet table with given category

\* It will also update count of words (or insert new) in wordFrequency table

\*/

public function train($sentence, $category) {

$spam = Category::$SPAM;

$ham = Category::$HAM;

if ($category == $spam || $category == $ham) {

//connecting to database

require 'db\_connect.php';

// inserting sentence into trainingSet with given category

$sql = mysqli\_query($conn, "INSERT into trainingSet (document, category) values('$sentence', '$category')");

// extracting keywords

$keywordsArray = $this -> tokenize($sentence);

// updating wordFrequency table

foreach ($keywordsArray as $word) {

// if this word is already present with given category then update count else insert

$sql = mysqli\_query($conn, "SELECT count(\*) as total FROM wordFrequency WHERE word = '$word' and category= '$category' ");

$count = mysqli\_fetch\_assoc($sql);

if ($count['total'] == 0) {

$sql = mysqli\_query($conn, "INSERT into wordFrequency (word, category, count) values('$word', '$category', 1)");

} else {

$sql = mysqli\_query($conn, "UPDATE wordFrequency set count = count + 1 where word = '$word'");

}

}

//closing connection

$conn -> close();

} else {

throw new Exception('Unknown category. Valid categories are: $ham, $spam');

}

}

/\*\*

\* this function takes a paragraph of text as input and returns an array of keywords.

\*/

private function tokenize($sentence) {

$stopWords = array('about','and','are','com','for','from','how',

'that','the','this', 'was','what','when','where','who','will','with','und','the','www');

//removing all the characters which ar not letters, numbers or space

$sentence = preg\_replace("/[^a-zA-Z 0-9]+/", "", $sentence);

//converting to lowercase

$sentence = strtolower($sentence);

//an empty array

$keywordsArray = array();

//splitting text into array of keywords

//http://www.w3schools.com/php/func\_string\_strtok.asp

$token = strtok($sentence, " ");

while ($token !== false) {

//excluding elements of length less than 3

if (!(strlen($token) <= 2)) {

//excluding elements which are present in stopWords array

//http://www.w3schools.com/php/func\_array\_in\_array.asp

if (!(in\_array($token, $stopWords))) {

array\_push($keywordsArray, $token);

}

}

$token = strtok(" ");

}

return $keywordsArray;

}

/\*\*

\* This function takes an array of words as input and return category (spam/ham) after

\* applying Naive Bayes Classifier

\*

\* Naive Bayes Classifier Algorithm -

\*

\* p(spam/bodyText) = p(spam) \* p(bodyText/spam) / p(bodyText);

\* p(ham/bodyText) = p(ham) \* p(bodyText/ham) / p(bodyText);

\* p(bodyText) is constant so it can be ommitted

\* p(spam) = no of documents (sentence) belonging to category spam / total no of documents (sentence)

\* p(bodyText/spam) = p(word1/spam) \* p(word2/spam) \* .... p(wordn/spam)

\* Laplace smoothing for such cases is usually given by (c+1)/(N+V),

\* where V is the vocabulary size (total no of different words)

\* p(word/spam) = no of times word occur in spam / no of all words in spam

\* Reference:

\* http://stackoverflow.com/questions/9996327/using-a-naive-bayes-classifier-to-classify-tweets-some-problems

\* https://github.com/ttezel/bayes/blob/master/lib/naive\_bayes.js

\*/

private function decide ($keywordsArray) {

$spam = Category::$SPAM;

$ham = Category::$HAM;

// by default assuming category to be ham

$category = $ham;

// making connection to database

require 'db\_connect.php';

$sql = mysqli\_query($conn, "SELECT count(\*) as total FROM trainingSet WHERE category = '$spam' ");

$spamCount = mysqli\_fetch\_assoc($sql);

$spamCount = $spamCount['total'];

$sql = mysqli\_query($conn, "SELECT count(\*) as total FROM trainingSet WHERE category = '$ham' ");

$hamCount = mysqli\_fetch\_assoc($sql);

$hamCount = $hamCount['total'];

$sql = mysqli\_query($conn, "SELECT count(\*) as total FROM trainingSet ");

$totalCount = mysqli\_fetch\_assoc($sql);

$totalCount = $totalCount['total'];

//p(spam)

$pSpam = $spamCount / $totalCount; // (no of documents classified as spam / total no of documents)

//p(ham)

$pHam = $hamCount / $totalCount; // (no of documents classified as ham / total no of documents)

//echo $pSpam." ".$pHam;

// no of distinct words (used for laplace smoothing)

$sql = mysqli\_query($conn, "SELECT count(\*) as total FROM wordFrequency ");

$distinctWords = mysqli\_fetch\_assoc($sql);

$distinctWords = $distinctWords['total'];

$bodyTextIsSpam = log($pSpam);

foreach ($keywordsArray as $word) {

$sql = mysqli\_query($conn, "SELECT count as total FROM wordFrequency where word = '$word' and category = '$spam' ");

$wordCount = mysqli\_fetch\_assoc($sql);

$wordCount = $wordCount['total'];

$bodyTextIsSpam += log(($wordCount + 1) / ($spamCount + $distinctWords));

}

$bodyTextIsHam = log($pHam);

foreach ($keywordsArray as $word) {

$sql = mysqli\_query($conn, "SELECT count as total FROM wordFrequency where word = '$word' and category = '$ham' ");

$wordCount = mysqli\_fetch\_assoc($sql);

$wordCount = $wordCount['total'];

$bodyTextIsHam += log(($wordCount + 1) / ($hamCount + $distinctWords));

}

if ($bodyTextIsHam >= $bodyTextIsSpam) {

$category = $ham;

} else {

$category = $spam;

}

$conn -> close();

return $category;

}

}

?>

P(spam|b1)

P(spam|t1t2…tx) = p(spam)\*P(t1|spam)