In [1]:

```
import nltk
```

In [ ]:

```
# nltk.download()
```

In [2]:

```
nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Santosh\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

Out[2]:

```
True
```

In [3]:

```
dir(nltk)
```

Out[3]:

```
['AbstractLazySequence',
 'AffixTagger',
 'AlignedSent',
 'Alignment',
 'AnnotationTask',
 'ApplicationExpression',
 'Assignment',
 'BigramAssocMeasures',
 'BigramCollocationFinder',
 'BigramTagger',
 'BinaryMaxentFeatureEncoding',
 'BlanklineTokenizer',
 'BllipParser',
 'BottomUpChartParser',
 'BottomUpLeftCornerChartParser',
 'BottomUpProbabilisticChartParser',
 'Boxer',
 'BrillTagger',
 'BrillTaggerTrainer',
 'CFG',
 'CRFTagger',
 'CfgReadingCommand',
 'ChartParser',
 'ChunkParserI',
 'ChunkScore',
 'Cistem',
 'ClassifierBasedPOSTagger',
 'ClassifierBasedTagger',
 'ClassifierI',
 'ConcordanceIndex',
 'ConditionalExponentialClassifier',
 'ConditionalFreqDist',
 'ConditionalProbDist',
 'ConditionalProbDistI',
 'ConfusionMatrix',
 'ContextIndex',
 'ContextTagger',
 'ContingencyMeasures',
 'CoreNLPDependencyParser',
 'CoreNLPParser',
 'Counter',
 'CrossValidationProbDist',
 'DRS',
 'DecisionTreeClassifier',
 'DefaultTagger',
 'DependencyEvaluator',
 'DependencyGrammar',
 'DependencyGraph',
 'DependencyProduction',
 'DictionaryConditionalProbDist',
 'DictionaryProbDist',
 'DiscourseTester',
 'DrtExpression',
 'DrtGlueReadingCommand',
 'ELEProbDist',
 'EarleyChartParser',
 'Expression',
 'FStructure',
 'FeatDict',
 'FeatList',
 'FeatStruct',
 'FeatStructReader',
 'Feature',
 'FeatureBottomUpChartParser',
 'FeatureBottomUpLeftCornerChartParser',
 'FeatureChartParser',
 'FeatureEarleyChartParser',
 'FeatureIncrementalBottomUpChartParser',
 'FeatureIncrementalBottomUpLeftCornerChartParser',
 'FeatureIncrementalChartParser',
 'FeatureIncrementalTopDownChartParser',
 'FeatureTopDownChartParser',
 'FreqDist',
 'HTTPPasswordMgrWithDefaultRealm',
 'HeldoutProbDist',
 'HiddenMarkovModelTagger',
 'HiddenMarkovModelTrainer',
 'HunposTagger',
 'IBMModel',
 'IBMModel1',
 'IBMModel2',
 'IBMModel3',
 'IBMModel4',
 'IBMModel5',
 'ISRIStemmer',
 'ImmutableMultiParentedTree',
 'ImmutableParentedTree',
 'ImmutableProbabilisticMixIn',
 'ImmutableProbabilisticTree',
 'ImmutableTree',
 'IncrementalBottomUpChartParser',
 'IncrementalBottomUpLeftCornerChartParser',
 'IncrementalChartParser',
 'IncrementalLeftCornerChartParser',
 'IncrementalTopDownChartParser',
 'Index',
 'InsideChartParser',
 'JSONTaggedDecoder',
 'JSONTaggedEncoder',
 'KneserNeyProbDist',
 'LancasterStemmer',
 'LaplaceProbDist',
 'LazyConcatenation',
 'LazyEnumerate',
 'LazyIteratorList',
 'LazyMap',
 'LazySubsequence',
 'LazyZip',
 'LeftCornerChartParser',
 'LidstoneProbDist',
 'LineTokenizer',
 'LogicalExpressionException',
 'LongestChartParser',
 'MLEProbDist',
 'MWETokenizer',
 'Mace',
 'MaceCommand',
 'MaltParser',
 'MaxentClassifier',
 'Model',
 'MultiClassifierI',
 'MultiParentedTree',
 'MutableProbDist',
 'NaiveBayesClassifier',
 'NaiveBayesDependencyScorer',
 'NgramAssocMeasures',
 'NgramTagger',
 'NonprojectiveDependencyParser',
 'Nonterminal',
 'OrderedDict',
 'PCFG',
 'Paice',
 'ParallelProverBuilder',
 'ParallelProverBuilderCommand',
 'ParentedTree',
 'ParserI',
 'PerceptronTagger',
 'PhraseTable',
 'PorterStemmer',
 'PositiveNaiveBayesClassifier',
 'ProbDistI',
 'ProbabilisticDependencyGrammar',
 'ProbabilisticMixIn',
 'ProbabilisticNonprojectiveParser',
 'ProbabilisticProduction',
 'ProbabilisticProjectiveDependencyParser',
 'ProbabilisticTree',
 'Production',
 'ProjectiveDependencyParser',
 'Prover9',
 'Prover9Command',
 'ProxyBasicAuthHandler',
 'ProxyDigestAuthHandler',
 'ProxyHandler',
 'PunktSentenceTokenizer',
 'QuadgramCollocationFinder',
 'RSLPStemmer',
 'RTEFeatureExtractor',
 'RUS_PICKLE',
 'RandomChartParser',
 'RangeFeature',
 'ReadingCommand',
 'RecursiveDescentParser',
 'RegexpChunkParser',
 'RegexpParser',
 'RegexpStemmer',
 'RegexpTagger',
 'RegexpTokenizer',
 'ReppTokenizer',
 'ResolutionProver',
 'ResolutionProverCommand',
 'SExprTokenizer',
 'SLASH',
 'Senna',
 'SennaChunkTagger',
 'SennaNERTagger',
 'SennaTagger',
 'SequentialBackoffTagger',
 'ShiftReduceParser',
 'SimpleGoodTuringProbDist',
 'SklearnClassifier',
 'SlashFeature',
 'SnowballStemmer',
 'SpaceTokenizer',
 'StackDecoder',
 'StanfordNERTagger',
 'StanfordPOSTagger',
 'StanfordSegmenter',
 'StanfordTagger',
 'StemmerI',
 'SteppingChartParser',
 'SteppingRecursiveDescentParser',
 'SteppingShiftReduceParser',
 'TYPE',
 'TabTokenizer',
 'TableauProver',
 'TableauProverCommand',
 'TaggerI',
 'TestGrammar',
 'Text',
 'TextCat',
 'TextCollection',
 'TextTilingTokenizer',
 'TnT',
 'TokenSearcher',
 'ToktokTokenizer',
 'TopDownChartParser',
 'TransitionParser',
 'Tree',
 'TreebankWordTokenizer',
 'Trie',
 'TrigramAssocMeasures',
 'TrigramCollocationFinder',
 'TrigramTagger',
 'TweetTokenizer',
 'TypedMaxentFeatureEncoding',
 'Undefined',
 'UniformProbDist',
 'UnigramTagger',
 'UnsortedChartParser',
 'Valuation',
 'Variable',
 'ViterbiParser',
 'WekaClassifier',
 'WhitespaceTokenizer',
 'WittenBellProbDist',
 'WordNetLemmatizer',
 'WordPunctTokenizer',
 '__author__',
 '__author_email__',
 '__builtins__',
 '__cached__',
 '__classifiers__',
 '__copyright__',
 '__doc__',
 '__file__',
 '__keywords__',
 '__license__',
 '__loader__',
 '__longdescr__',
 '__maintainer__',
 '__maintainer_email__',
 '__name__',
 '__package__',
 '__path__',
 '__spec__',
 '__url__',
 '__version__',
 'absolute_import',
 'accuracy',
 'add_logs',
 'agreement',
 'align',
 'alignment_error_rate',
 'aline',
 'api',
 'app',
 'apply_features',
 'approxrand',
 'arity',
 'association',
 'bigrams',
 'binary_distance',
 'binary_search_file',
 'binding_ops',
 'bisect',
 'blankline_tokenize',
 'bleu',
 'bleu_score',
 'bllip',
 'boolean_ops',
 'boxer',
 'bracket_parse',
 'breadth_first',
 'brill',
 'brill_trainer',
 'build_opener',
 'call_megam',
 'casual',
 'casual_tokenize',
 'ccg',
 'chain',
 'chart',
 'chat',
 'choose',
 'chunk',
 'cistem',
 'class_types',
 'classify',
 'clause',
 'clean_html',
 'clean_url',
 'cluster',
 'collections',
 'collocations',
 'combinations',
 'compat',
 'config_java',
 'config_megam',
 'config_weka',
 'conflicts',
 'confusionmatrix',
 'conllstr2tree',
 'conlltags2tree',
 'corenlp',
 'corpus',
 'crf',
 'custom_distance',
 'data',
 'decisiontree',
 'decorator',
 'decorators',
 'defaultdict',
 'demo',
 'dependencygraph',
 'deque',
 'discourse',
 'distance',
 'download',
 'download_gui',
 'download_shell',
 'downloader',
 'draw',
 'drt',
 'earleychart',
 'edit_distance',
 'elementtree_indent',
 'entropy',
 'equality_preds',
 'evaluate',
 'evaluate_sents',
 'everygrams',
 'extract_rels',
 'extract_test_sentences',
 'f_measure',
 'featstruct',
 'featurechart',
 'filestring',
 'find',
 'flatten',
 'fractional_presence',
 'getproxies',
 'ghd',
 'glue',
 'grammar',
 'guess_encoding',
 'help',
 'hmm',
 'hunpos',
 'ibm1',
 'ibm2',
 'ibm3',
 'ibm4',
 'ibm5',
 'ibm_model',
 'ieerstr2tree',
 'improved_close_quote_regex',
 'improved_open_quote_regex',
 'improved_open_single_quote_regex',
 'improved_punct_regex',
 'in_idle',
 'induce_pcfg',
 'inference',
 'infile',
 'inspect',
 'install_opener',
 'internals',
 'interpret_sents',
 'interval_distance',
 'invert_dict',
 'invert_graph',
 'is_rel',
 'islice',
 'isri',
 'jaccard_distance',
 'json_tags',
 'jsontags',
 'lancaster',
 'lazyimport',
 'lfg',
 'line_tokenize',
 'linearlogic',
 'load',
 'load_parser',
 'locale',
 'log_likelihood',
 'logic',
 'mace',
 'malt',
 'map_tag',
 'mapping',
 'masi_distance',
 'maxent',
 'megam',
 'memoize',
 'metrics',
 'misc',
 'mwe',
 'naivebayes',
 'ne_chunk',
 'ne_chunk_sents',
 'ngrams',
 'nonprojectivedependencyparser',
 'nonterminals',
 'numpy',
 'os',
 'pad_sequence',
 'paice',
 'parse',
 'parse_sents',
 'pchart',
 'perceptron',
 'pk',
 'porter',
 'pos_tag',
 'pos_tag_sents',
 'positivenaivebayes',
 'pprint',
 'pr',
 'precision',
 'presence',
 'print_function',
 'print_string',
 'probability',
 'projectivedependencyparser',
 'prover9',
 'punkt',
 'py25',
 'py26',
 'py27',
 'pydoc',
 'python_2_unicode_compatible',
 'raise_unorderable_types',
 'ranks_from_scores',
 'ranks_from_sequence',
 're',
 're_show',
 'read_grammar',
 'read_logic',
 'read_valuation',
 'recall',
 'recursivedescent',
 'regexp',
 'regexp_span_tokenize',
 'regexp_tokenize',
 'register_tag',
 'relextract',
 'repp',
 'resolution',
 'ribes',
 'ribes_score',
 'root_semrep',
 'rslp',
 'rte_classifier',
 'rte_classify',
 'rte_features',
 'rtuple',
 'scikitlearn',
 'scores',
 'segmentation',
 'sem',
 'senna',
 'sent_tokenize',
 'sequential',
 'set2rel',
 'set_proxy',
 'sexpr',
 'sexpr_tokenize',
 'shiftreduce',
 'simple',
 'sinica_parse',
 'skipgrams',
 'skolemize',
 'slice_bounds',
 'snowball',
 'spearman',
 'spearman_correlation',
 'stack_decoder',
 'stanford',
 'stanford_segmenter',
 'stem',
 'str2tuple',
 'string_span_tokenize',
 'string_types',
 'subprocess',
 'subsumes',
 'sum_logs',
 'sys',
 'tableau',
 'tadm',
 'tag',
 'tagset_mapping',
 'tagstr2tree',
 'tbl',
 'text',
 'text_type',
 'textcat',
 'texttiling',
 'textwrap',
 'tkinter',
 'tnt',
 'tokenize',
 'tokenwrap',
 'toktok',
 'toolbox',
 'total_ordering',
 'total_ordering',
 'transitionparser',
 'transitive_closure',
 'translate',
 'tree',
 'tree2conllstr',
 'tree2conlltags',
 'treebank',
 'treetransforms',
 'trigrams',
 'tuple2str',
 'types',
 'unify',
 'unique_list',
 'untag',
 'usage',
 'util',
 'version_file',
 'version_info',
 'viterbi',
 'weka',
 'windowdiff',
 'word_tokenize',
 'wordnet',
 'wordpunct_tokenize',
 'wsd']
```

tokenize example

In [4]:

```
from nltk.tokenize import word_tokenize
input_text='I am learning NLP usink nltk'
word_token=word_tokenize(input_text)
print(input_text)
print(word_token)
```

```
I am learning NLP usink nltk
['I', 'am', 'learning', 'NLP', 'usink', 'nltk']
```

Reading a Text Data

Method 1: Using open()

Method 2: pandas read_csv()

In [5]:

```
# Method 1: Using open()
raw_data=open("SMSSpamCollection").read()
```

In [6]:

```
raw_data[0:500]
```

Out[6]:

```
"ham\tGo until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine
there got amore wat...\nham\tOk lar... Joking wif u oni...\nspam\tFree entry in 2 a wkly comp to w
in FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's
apply 08452810075over18's\nham\tU dun say so early hor... U c already then say...\nham\tNah I don'
t think he goes to usf, he lives around here though\nspam\tFreeMsg Hey there darling it's been 3 w
eek's now and no word bac"
```

In [7]:

```
parsed_data=raw_data.replace('\t', '\n').split('\n')
parsed_data[0:10]
```

```
parsed_data[0:10]
```

Out[7]:

```
['ham',
 'Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g
ot amore wat...',
 'ham',
 'Ok lar... Joking wif u oni...',
 'spam',
 "Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive
entry question(std txt rate)T&C's apply 08452810075over18's",
 'ham',
 'U dun say so early hor... U c already then say...',
 'ham',
 "Nah I don't think he goes to usf, he lives around here though"]
```

In [8]:

```
label_list=parsed_data[0::2]
msg_list=parsed_data[1::2]
print(label_list[0:5])
print(msg_list[0:5])
```

```
['ham', 'ham', 'spam', 'ham', 'ham']
['Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g
ot amore wat...', 'Ok lar... Joking wif u oni...', "Free entry in 2 a wkly comp to win FA Cup fina
l tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply
08452810075over18's", 'U dun say so early hor... U c already then say...', "Nah I don't think he g
oes to usf, he lives around here though"]
```

In [9]:

```
print(len(label_list))
print(len(msg_list))
```

```
5575
5574
```

In [10]:

```
print(msg_list[-3:])
print(label_list[-3:])
```

```
['Pity, * was in mood for that. So...any other suggestions?', "The guy did some bitching but I act
ed like i'd be interested in buying something else next week and he gave it to us for free", 'Rofl
. Its true to its name']
['ham', 'ham', '']
```

In [11]:

```
import pandas as pd
combined_df=pd.DataFrame({'label':label_list[:-1],'sms':msg_list})
combined_df.head()
```

Out[11]:

	label	sms
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

Method 2: pandas read_csv

In [12]:

```
# Method 2: pandas read_csv
dataset=pd.read_csv("SMSSpamCollection", sep='\t', header=None, names=['label', 'sms'])
```

In [13]:

```
dataset.head()
```

Out[13]:

	label	sms
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

Exploring the data

In [14]:

```
# Shape of the dataset
dataset.shape
```

Out[14]:

```
(5572, 2)
```

In [15]:

```
# Number of labeles: ham and spam, in ratio
#count
print('\n count:', '\n',dataset.label.value_counts())
print('\n')
#ratio
print('ratio : ', '\n',dataset.label.value_counts()/dataset.shape[0])
```

```
 count:
 ham     4825
spam     747
Name: label, dtype: int64


ratio :
 ham     0.865937
spam    0.134063
Name: label, dtype: float64
```

In [16]:

```
# Number of missing values
dataset.apply(lambda x: x.isnull().sum())
```

Out[16]:

```
label    0
sms      0
dtype: int64
```

NLP Pipeline

NLP Pipeline

Raw Data===> Tokenization ===> Text Cleaning ===> Vectorization===> ML ALgorithm ===> Evaluate Model ==> Deploy

Text Preprocessing: Tokenization + Text Cleaning

- Remove Punctuation
- Tokenization
- Remove Stop words
- Stemming

Remove Punctuation

In [18]:

```
import pandas as pd
pd.set_option('display.max_colwidth', 100)
data=pd.read_csv('SMSSpamCollection', sep='\t', header=None, names=['label', 'msg'])
data.head()
```

Out[18]:

	label	msg
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives around here though

In [19]:

```
import string
string.punctuation
```

Out[19]:

```
'!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
```

In [23]:

```
def remove_punctuation(txt):
    tex_nopunct=''.join([c for c in txt if c not in string.punctuation])
    return tex_nopunct
```

In [24]:

```
data['msg_clean']=data['msg'].apply(lambda x: remove_punctuation(x))
data.head()
```

Out[24]:

	label	msg	msg_clean
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	Go until jurong point crazy Available only in bugis n great world la e buffet Cine there got amo...
1	ham	Ok lar... Joking wif u oni...	Ok lar Joking wif u oni
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005 Text FA to 87121 to receive e...
3	ham	U dun say so early hor... U c already then say...	U dun say so early hor U c already then say
4	ham	Nah I don't think he goes to usf, he lives around here though	Nah I dont think he goes to usf he lives around here though

Tokenization

- re.split(w)==> split on word characters
- re.split(W)==> split on non word characters
- re.split(W+)==> split on one or more non word characters

In [25]:

```
import re

def tokenize(txt):

    tokens=re.split('\W+', txt)
    return tokens
```

In [27]:

```
data['msg_clean_tokens']=data['msg_clean'].apply(lambda x: tokenize(x.lower()))
data.head()
```

Out[27]:

	label	msg	msg_clean	msg_clean_tokens
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	Go until jurong point crazy Available only in bugis n great world la e buffet Cine there got amo...	[go, until, jurong, point, crazy, available, only, in, bugis, n, great, world, la, e, buffet, ci...
1	ham	Ok lar... Joking wif u oni...	Ok lar Joking wif u oni	[ok, lar, joking, wif, u, oni]
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005 Text FA to 87121 to receive e...	[free, entry, in, 2, a, wkly, comp, to, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, to...
3	ham	U dun say so early hor... U c already then say...	U dun say so early hor U c already then say	[u, dun, say, so, early, hor, u, c, already, then, say]
4	ham	Nah I don't think he goes to usf, he lives around here though	Nah I dont think he goes to usf he lives around here though	[nah, i, dont, think, he, goes, to, usf, he, lives, around, here, though]

Removing Stop Words

In [34]:

```
import nltk
stopwords=nltk.corpus.stopwords.words('english')
stopwords[0:10]
```

Out[34]:

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're"]
```

In [35]:

```
def remove_stopwords(txt):

    txt_clean=[c for c in txt if c not in stopwords]
    return txt_clean
```

In [38]:

```
data['msg_no_stop']=data['msg_clean_tokens'].apply(lambda x: remove_stopwords(x))
data.head()
```

Out[38]:

	label	msg	msg_clean	msg_clean_tokens	msg_no_stop
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	Go until jurong point crazy Available only in bugis n great world la e buffet Cine there got amo...	[go, until, jurong, point, crazy, available, only, in, bugis, n, great, world, la, e, buffet, ci...	[go, jurong, point, crazy, available, bugis, n, great, world, la, e, buffet, cine, got, amore, wat]
1	ham	Ok lar... Joking wif u oni...	Ok lar Joking wif u oni	[ok, lar, joking, wif, u, oni]	[ok, lar, joking, wif, u, oni]
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005 Text FA to 87121 to receive e...	[free, entry, in, 2, a, wkly, comp, to, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, to...	[free, entry, 2, wkly, comp, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, 87121, receiv...
3	ham	U dun say so early hor... U c already then say...	U dun say so early hor U c already then say	[u, dun, say, so, early, hor, u, c, already, then, say]	[u, dun, say, early, hor, u, c, already, say]
4	ham	Nah I don't think he goes to usf, he lives around here though	Nah I dont think he goes to usf he lives around here though	[nah, i, dont, think, he, goes, to, usf, he, lives, around, here, though]	[nah, dont, think, goes, usf, lives, around, though]

Stemming

- Process of reducing derived words to their root word
- Eg: coder, coding, coded===> Code (root word)

Errors in Stemming

1. Over Stemming

   - too much of word is cut off(meaning is lost)
   - 2 words of different stem reduced to same stem

2. Under Stemming

   - 2 words of same stem mapped to different stems

why is stemming useful

- Reduces the corpus of words the model needs to work with
- Explicitly correlates the words with similar meaning

Stemming algorithms

1. Porter Stemmer-------> more popular
2. Snowball Stemmer
3. Lancaster Stemmer
4. Regex-based Stemmer

Porter Stemmer

In [40]:

```
import nltk
from nltk.stem import PorterStemmer
ps=PorterStemmer()
```

In [41]:

```
dir(ps)
```

Out[41]:

```
['MARTIN_EXTENSIONS',
 'NLTK_EXTENSIONS',
 'ORIGINAL_ALGORITHM',
 '__abstractmethods__',
 '__class__',
 '__delattr__',
 '__dict__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__le__',
 '__lt__',
 '__module__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 '__unicode__',
 '__weakref__',
 '_abc_impl',
 '_apply_rule_list',
 '_contains_vowel',
 '_ends_cvc',
 '_ends_double_consonant',
 '_has_positive_measure',
 '_is_consonant',
 '_measure',
 '_replace_suffix',
 '_step1a',
 '_step1b',
 '_step1c',
 '_step2',
 '_step3',
 '_step4',
 '_step5a',
 '_step5b',
 'mode',
 'pool',
 'stem',
 'unicode_repr',
 'vowels']
```

- we are interested in ps.stem

In [42]:

```
print(ps.stem('coder'))
print(ps.stem('coded'))
print(ps.stem('coding'))
print(ps.stem('code'))
```

```
coder
code
code
code
```

In [43]:

```
print(ps.stem('data'))
print(ps.stem('datum'))
```

```
data
datum
```

- ==> data and datum are same with meaning ==> under stemming

In [44]:

```
## SMSspam Cleaning== > all in one
```

In [45]:

```
import nltk
import string
import re

stopwords=nltk.corpus.stopwords.words('english')
pd.set_option('display.max_colwidth', 100)
```

In [47]:

```
data=pd.read_csv('SMSSpamCollection', sep='\t', header=None, names=['label', 'msg'])
data.head()
```

Out[47]:

	label	msg
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives around here though

Clean Text

In [52]:

```
def clean_text(txt):
    text=''.join([c for c in txt if c not in string.punctuation])
    tokens=re.split('\W+', text)
    text_nostop=[c for c in tokens if c not in stopwords]
    return text_nostop
```

In [54]:

```
data['msg_nostop']=data['msg'].apply(lambda x: clean_text(x.lower()))
data.head()
```

Out[54]:

	label	msg	msg_nostop
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	[go, jurong, point, crazy, available, bugis, n, great, world, la, e, buffet, cine, got, amore, wat]
1	ham	Ok lar... Joking wif u oni...	[ok, lar, joking, wif, u, oni]
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	[free, entry, 2, wkly, comp, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, 87121, receiv...
3	ham	U dun say so early hor... U c already then say...	[u, dun, say, early, hor, u, c, already, say]
4	ham	Nah I don't think he goes to usf, he lives around here though	[nah, dont, think, goes, usf, lives, around, though]

Stem the Text

In [55]:

```
def stemming(tokenized_text):
    text=[ps.stem(word) for word in tokenized_text]
    return text
```

In [56]:

```
data['msg_stemmed']=data['msg_nostop'].apply(lambda x: stemming(x))
data.head()
```

Out[56]:

	label	msg	msg_nostop	msg_stemmed
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	[go, jurong, point, crazy, available, bugis, n, great, world, la, e, buffet, cine, got, amore, wat]	[go, jurong, point, crazi, avail, bugi, n, great, world, la, e, buffet, cine, got, amor, wat]
1	ham	Ok lar... Joking wif u oni...	[ok, lar, joking, wif, u, oni]	[ok, lar, joke, wif, u, oni]
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	[free, entry, 2, wkly, comp, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, 87121, receiv...	[free, entri, 2, wkli, comp, win, fa, cup, final, tkt, 21st, may, 2005, text, fa, 87121, receiv,...
3	ham	U dun say so early hor... U c already then say...	[u, dun, say, early, hor, u, c, already, say]	[u, dun, say, earli, hor, u, c, alreadi, say]
4	ham	Nah I don't think he goes to usf, he lives around here though	[nah, dont, think, goes, usf, lives, around, though]	[nah, dont, think, goe, usf, live, around, though]

Lemmatization

- Process of grouping together the inflected forms of a word to be analyzed as a single root word or lemma
- Unlike Stemming it reduces the inflected(derived) words properly ensuring that the root word(lemma) belongs to the language
- A lemma is the canonical form, dictionary form or citation form of a set of words
- Eg: bowl, bowled, bowling------> bowl(lemma, root word)
- does vocabulary analysis of words
- Slower than stemming but it is more accurate

Lemmatization vs Stemming

- speed vs accuracy tradeoff

Stemming: is typically faster

```
* simply chops off the end of a word using heuristics
* no understtanding of the context
```

Lemmatization: is typically more accurate

```
* Uses more informed analysis
* Always Reduces to a dictionary word
* More Acurate but computationally expensive
```

In [75]:

```
import nltk
nltk.download('wordnet')
wn=nltk.WordNetLemmatizer()
ps=PorterStemmer()
```

```
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\Santosh\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping corpora\wordnet.zip.
```

In [76]:

```
dir(wn)
```

Out[76]:

```
['__class__',
 '__delattr__',
 '__dict__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__le__',
 '__lt__',
 '__module__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 '__unicode__',
 '__weakref__',
 'lemmatize',
 'unicode_repr']
```

- we are interested in wn.lemmatize() function

In [77]:

```
print(ps.stem('goose'))
print(ps.stem('geese'))
```

```
goos
gees
```

In [78]:

```
wn=nltk.WordNetLemmatizer()
print(wn.lemmatize('goose'))
print(wn.lemmatize('geese'))
```

```
goose
goose
```

In [80]:

```
print(ps.stem('cactus'))
print(ps.stem('cacti'))
```

```
cactu
cacti
```

In [79]:

```
print(wn.lemmatize('cactus'))
print(wn.lemmatize('cacti'))
```

```
cactus
cactus
```

Read Raw Text

In [86]:

```
import nltk
import re
import pandas as pd
import string
stopwords=nltk.corpus.stopwords.words('english')

pd.set_option('display.max_colwidth', 100)
data=pd.read_csv('SMSSpamCollection', sep='\t',header=None, names=['label', 'msg'])
data.head()
```

Out[86]:

	label	msg
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives around here though

In [89]:

```
### text cleaning
def clean_text(txt):
    text=''.join([c for c in txt if c not in string.punctuation])
    tokens=re.split('\W+', text)
    text_nostop=[c for c in tokens if c not in stopwords]
    return text_nostop
```

In [91]:

```
data['msg_nostop']=data['msg'].apply(lambda x: clean_text(x.lower()))
data.head()
```

Out[91]:

	label	msg	msg_nostop
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	[go, jurong, point, crazy, available, bugis, n, great, world, la, e, buffet, cine, got, amore, wat]
1	ham	Ok lar... Joking wif u oni...	[ok, lar, joking, wif, u, oni]
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	[free, entry, 2, wkly, comp, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, 87121, receiv...
3	ham	U dun say so early hor... U c already then say...	[u, dun, say, early, hor, u, c, already, say]
4	ham	Nah I don't think he goes to usf, he lives around here though	[nah, dont, think, goes, usf, lives, around, though]

In [95]:

```
def lemmatization(token_text):
    text=[wn.lemmatize(word) for word in token_text]
    return text
```

In [96]:

```
data['msg_lemmatized']=data['msg_nostop'].apply(lambda x: lemmatization(x))
data.head()
```

Out[96]:

	label	msg	msg_nostop	msg_lemmatized

	label	msg	msg_nostop	msg_lemmatized
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	[go, jurong, point, crazy, available, bugis, n, great, world, la, e, buffet, cine, got, amore, wat]	[go, jurong, point, crazy, available, bugis, n, great, world, la, e, buffet, cine, got, amore, wat]
1	ham	Ok lar... Joking wif u oni...	[ok, lar, joking, wif, u, oni]	[ok, lar, joking, wif, u, oni]
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	[free, entry, 2, wkly, comp, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, 87121, receiv...	[free, entry, 2, wkly, comp, win, fa, cup, final, tkts, 21st, may, 2005, text, fa, 87121, receiv...
3	ham	U dun say so early hor... U c already then say...	[u, dun, say, early, hor, u, c, already, say]	[u, dun, say, early, hor, u, c, already, say]
4	ham	Nah I don't think he goes to usf, he lives around here though	[nah, dont, think, goes, usf, lives, around, though]	[nah, dont, think, go, usf, life, around, though]

Vectorization

- Process of encoding text as Feature Vectors
- **Feature Vector** : Vector of numerical features that represent an object

- Example of How CountVectorizer Works
  - w1 w2 w3 ...... w100 | label
  - 0 0 2 ...... 3 | 0(ham)
  - 4 0 1 ...... 0 | 1(spam)
- ==> this is **Document Matrix** or Document term matrix

Types of Vectorization

- Count Vectorization
- N-grams
- TF-IDF

Count Vectorization

- creates document term matrix

from sklearn.feature_extraction.text import CountVectorizer

cv=CountVectorizer(analyzer=)

Read Raw Text

In [106]:

```
import pandas as pd
import re
import string
import nltk

stopwords=nltk.corpus.stopwords.words('english')
ps=PorterStemmer()

data=pd.read_csv("SMSSpamCollection", sep='\t', header=None, names=['label', 'msg'])
data.head()
```

Out[106]:

	label	msg
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...

	label	msg
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives around here though

Clean Text

In [120]:

```
def clean_text(txt):
    text=''.join([c for c in txt if c not in string.punctuation])
    tokens=re.split('\W+', text)
    txt_nostop=[ps.stem(word) for word in tokens if word not in stopwords]
    return txt_nostop
```

CountVectorizer Example

In [110]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer()

corpus=['This is a sentence is',
       'this is another sentence',
       'third document is here']

X=cv.fit(corpus)
print(X.vocabulary_)
print(cv.get_feature_names())
```

```
{'this': 6, 'is': 3, 'sentence': 4, 'another': 0, 'third': 5, 'document': 1, 'here': 2}
['another', 'document', 'here', 'is', 'sentence', 'third', 'this']
```

In [113]:

```
X=cv.transform(corpus)

print('X shape :', X.shape)
print('\n')
print('X:',X)
print('\n')

print(X.toarray())
```

```
X shape : (3, 7)


X:   (0, 3)	2
  (0, 4)	1
  (0, 6)	1
  (1, 0)	1
  (1, 3)	1
  (1, 4)	1
  (1, 6)	1
  (2, 1)	1
  (2, 2)	1
  (2, 3)	1
  (2, 5)	1


[[0 0 0 2 1 0 1]
 [1 0 0 1 1 0 1]
 [0 1 1 1 0 1 0]]
```

In [117]:

```
df=pd.DataFrame(X.toarray(), columns=cv.get_feature_names())
df
```

Out[117]:

	another	document	here	is	sentence	third	this
0	0	0	0	2	1	0	1
1	1	0	0	1	1	0	1
2	0	1	1	1	0	1	0

CountVectorization on SMSspamCollections

In [121]:

```
cv1=CountVectorizer(analyzer=clean_text)
X=cv1.fit_transform(data['msg'])
X.shape
```

Out[121]:

```
(5572, 8340)
```

In [123]:

```
print(cv1.get_feature_names())
```

```
['', '0', '008704050406', '0089mi', '0121', '01223585236', '01223585334', '0125698789', '02',
'020603', '0207', '02070836089', '02072069400', '02073162414', '02085076972', '020903', '021', '05
0703', '0578', '06', '060505', '061104', '07008009200', '07046744435', '07090201529',
'07090298926', '07099833605', '071104', '07123456789', '0721072', '07732584351', '07734396839', '0
7742676969', '07753741225', '0776xxxxxxx', '07786200117', '077xxx', '078', '07801543489', '07808',
'07808247860', '07808726822', '07815296484', '07821230901', '0784987', '0789xxxxxxx',
'0794674629107880867867', '0796xxxxxx', '07973788240', '07xxxxxxxxx', '0800', '08000407165',
'08000776320', '08000839402', '08000930705', '08000938767', '08001950382', '08002888812',
'08002986030', '08002986906', '08002988890', '08006344447', '0808', '08081263000', '08081560665',
'0825', '0844', '08448350055', '08448714184', '0845', '08450542832', '08452810071', '08452810073',
'08452810075over18', '0870', '08700621170150p', '08701213186', '08701237397', '08701417012',
'08701417012150p', '0870141701216', '087016248', '08701752560', '087018728737', '0870241182716', '
08702490080', '08702840625', '08702840625comuk', '08704439680', '08704439680tsc', '08706091795', '
0870737910216yr', '08707500020', '08707509020', '0870753331018', '08707808226', '08708034412', '08
708800282', '08709222922', '08709501522', '0870k', '087104711148', '08712101358', '08712103738', '
0871212025016', '08712300220', '087123002209am7pm', '08712317606', '08712400200', '08712400603', '
08712402050', '08712402578', '08712402779', '08712402902', '08712402972', '08712404000',
'08712405020', '08712405022', '08712460324', '08712460324nat', '08712466669', '0871277810710pmin',
'0871277810810', '0871277810910pmin', '087143423992stop', '087147123779am7pm', '08714712379',
'08714712388', '08714712394', '08714712412', '08714714011', '08714719523', '08715203028',
'08715203649', '08715203652', '08715203656', '08715203677', '08715203685', '08715203694',
'08715205273', '08715500022', '08715705022', '08717111821', '08717168528', '08717205546',
'08717507382', '08717507711', '08717509990', '08717890890', '08717895698', '08717898035',
'08718711108', '08718720201', '08718723815', '08718725756', '08718726270',
'08718726270150gbpmtmsg18', '08718726970', '08718726971', '08718726978', '087187272008',
'08718727868', '08718727870', '08718729755', '08718729758', '08718730555', '08718730666',
'08718738001', '08718738002', '08718738034', '08719180219', '08719180248', '08719181259',
'08719181503', '08719181513', '08719839835', '08719899217', '08719899229', '08719899230',
'09041940223', '09050000301', '09050000332', '09050000460', '09050000555', '09050000878',
'09050000928', '09050001295', '09050001808', '09050002311', '09050003091', '09050005321',
'09050090044', '09050280520', '09053750005', '09056242159', '09057039994', '09058091854',
'09058091870', '09058094454', '09058094455', '09058094507', '09058094565', '09058094583',
'09058094594', '09058094597', '09058094599', '09058095107', '09058095201', '09058097189',
'09058097218', '09058098002', '09058099801', '09061104276', '09061104283', '09061209465',
'09061213237', '09061221061', '09061221066', '09061701444', '09061701461', '09061701851',
'09061701939', '09061702893', '09061743386', '09061743806', '09061743810', '09061743811',
'09061744553', '09061749602', '09061790121', '09061790125', '09061790126', '09063440451',
'09063442151', '09063458130', '0906346330', '09064011000', '09064012103', '09064012160',
'09064015307', '09064017295', '09064017305', '09064018838', '09064019014', '09064019788',
'09065069120', '09065069154', '09065171142stopsms08', '09065171142stopsms08718727870150ppm',
'09065174042', '09065394514', '09065394973', '09065989180', '09065989182', '09066350750',
'09066358152', '09066358361', '09066361921', '09066362206', '09066362220', '09066362231',
'09066364311', '09066364349', '09066364589', '09066368327', '09066368470', '09066368753',
'09066380611', '09066382422', '09066612661', '09066649731from', '09066660100', '09071512432',
'09071512433', '09071517866', '09077818151', '09090204448', '09090900040', '09094100151',
'09094646631', '09094646899', '09095350301', '09096102316', '09099725823', '09099726395',
'09099726429', '09099726481', '09099726553', '09111030116', '09111032124', '09701213186',
'0anetwork', '1', '10', '100', '1000', '10000', '100000', '1000call', '100603', '100psm', '1010'
```

```
'0anetwork', '1', '10', '100', '1000', '10000', '100000', '1000call', '100603', '100psm', '1010',
'1013', '101mega', '1030', '10803', '10am', '10am7pm', '10am9pm', '10k', '10p', '10pmin', '10ppm',
'10th', '11', '1120', '113', '1131', '11414', '1146', '1148', '116', '1172', '118pmsg', '11mth', '
12', '120', '12000pe', '1205', '121', '1225', '123', '1230', '125', '1250', '125gift', '128', '12h
our', '12hr', '12mth', '12price', '13', '130', '131004', '1327', '13404', '139', '140', '1405', '1
40ppm', '145', '1450', '146tf150p', '14thmarch', '150', '1500', '150ea', '150morefrmmob',
'150msg', '150mtmsgrcvd18', '150p', '150pday', '150perweeksub', '150perwksub', '150pm', '150pmeg',
'150pmin', '150pmmorefrommobile2bremovedmobypobox734ls27yf', '150pmsg', '150pmsgrcvd',
'150pmsgrcvdhgsuite3422landsroww1j6hl', '150pmt', '150pmtmsg', '150pmtmsgrcvd18',
'150ppermesssubscript', '150ppm', '150ppmpobox10183bhamb64x', '150ppmsg', '150prcvd', '150psm', '1
50ptext', '150ptone', '150pw', '150pwk', '150rcvd', '150week', '150wk', '151', '1526', '153', '155
41', '15pmin', '16', '1680', '169', '16onli', '177', '18', '180', '181104', '1843', '186',
'18onli', '18ptxt', '18yr', '195', '1956669', '1U', '1appledayno', '1childish', '1cup', '1da', '1e
r', '1hanuman', '1hi', '1hr', '1im', '1lemondayno', '1mcflyall', '1million', '1minmobsmor', '1minm
obsmorelkpobox177hp51fl', '1minmoremobsemspobox45po139wa', '1month', '1pm', '1s', '1st',
'1st4term', '1stchoicecouk', '1stone', '1tulsi', '1u', '1unbreak', '1winaweek', '1winawk',
'1x150pwk', '1yf', '2', '20', '200', '2000', '20000', '2003', '2004', '2005', '2006', '2007',
'2025050', '20f', '20m12aq', '20p', '20pmin', '21', '211104', '215', '21870000hi', '21m', '21st',
'22', '220cm2', '23', '2309', '230ish', '24', '241', '241004', '247mp', '24hr', '24m', '24th', '25
', '250', '250k', '255', '25f', '25p', '260305', '261004', '261104', '2667', '26th', '2703', '2760
3', '28', '2814032', '285', '28day', '28th', '28thfebtc', '290305', '29100', '29m', '2B', '2C', '2
I', '2U', '2bajarangabali', '2bold', '2channel', '2day', '2daylov', '2docdpleas', '2end', '2exit',
'2ez', '2getha', '2geva', '2go', '2godid', '2gthr', '2hook', '2hr', '2im', '2kbsubject',
'2marrow', '2moro', '2morow', '2morro', '2morrow', '2morrowxxxx', '2mro', '2mrw', '2mwen',
'2naughti', '2nd', '2nhite', '2night', '2nite', '2nitetel', '2optout', '2optoutd3wv', '2p',
'2polic', '2px', '2rcv', '2stop', '2stoptx', '2stoptxt', '2u', '2u2', '2untam', '2watershd', '2wax
sto', '2when', '2wk', '2wt', '2wu', '2year', '2yr', '3', '30', '300', '3000', '300603',
'300603tcsbcm4235wc1n3xxcallcost150ppmmobilesvari', '300p', '3030', '30apr', '30pptxt', '30th', '3
1', '3100', '310303', '311004', '31pmsg150p', '32000', '3230', '32323', '326', '32f', '330', '3350
', '3365', '350', '3510i', '35p', '3650', '36504', '3680', '3680offer', '373', '3750', '375max', '
38', '391784', '399', '3G', '3U', '3aj', '3cover', '3d', '3day', '3db', '3g', '3gbp', '3hr',
'3lion', '3lp', '3maruti', '3mile', '3min', '3mobil', '3optic', '3pound', '3qxj9', '3rd',
'3sentiment', '3ss', '3u', '3unkempt', '3uz', '3wife', '3wk', '3x', '3xx', '4', '40', '400', '400m
inscal', '402', '4041', '40411', '40533', '40gb', '40mph', '415', '41685', '41782', '420', '42049'
, '4217', '42478', '42810', '430', '434', '44', '4403ldnw1a7rw18', '447797706009', '447801259231',
'447per', '448712404000pleas', '449050000301', '449071512431', '449month', '45', '450', '450p', '4
50ppw', '450pw', '45239', '46', '47', '4712', '4742', '48', '4882', '48922', '49557', '4U', '4a',
'4brekki', '4cook', '4d', '4eva', '4few', '4fil', '4get', '4give', '4got', '4goten', '4info', '4jx
', '4lux', '4mi', '4mth', '4o', '4pavanaputra', '4press', '4rowdi', '4some1', '4tctxt', '4th', '4t
he', '4thnovbehind', '4txt120p', '4txtú120', '4u', '4ui', '4utxt', '4w', '4ward', '4wrd', '4year',
'5', '50', '500', '5000', '500000', '505060', '50award', '50p', '515', '515pm', '5226', '5249', '5
26', '528', '530', '532', '54', '542', '545', '5903', '5I', '5K', '5digit', '5free', '5ful', '5gar
den', '5gentli', '5ish', '5min', '5ml', '5month', '5p', '5pm', '5sankatmochan', '5terror', '5th',
'5wb', '5we', '5wkg', '5wq', '5year', '6', '600', '6031', '60400thousadi', '60p', '60pmin', '61200
', '61610', '62220cncl', '6230', '62468', '62735', '630', '63mile', '645', '645pm', '650', '6669',
'67441233', '68866', '69101', '69200', '69669', '69696', '69698', '69855', '6986618', '69876', '69
888', '69888nyt', '69911', '69969', '69988', '6cruel', '6day', '6hl', '6housemaid', '6hr', '6ish',
'6miss', '6month', '6pm', '6ramaduth', '6romant', '6th', '6time', '6wu', '6zf', '7', '700', '71',
'725', '7250', '7250i', '730', '730ish', '730pm', '731', '74355', '750', '75000', '7548', '7634',
'7684', '7732584351', '78', '786', '7876150ppm', '78pmin', '79', '7am', '7cfca1a', '7children',
'7ish', '7mahav', '7oz', '7pm', '7romant', '7shi', '7th', '7w', '7z', '8', '80', '800',
'8000930705', '80062', '8007', '80082', '80086', '80122300pwk', '80155', '80160', '80182', '8027',
'80488', '80488biz', '80608', '8077', '80878', '81010', '81151', '81303', '81618', '816183', '8224
2', '82277', '82277unsub', '82324', '82468', '830', '83021', '83039', '83049', '83110', '83118', '
83222', '83332pleas', '83338', '83355', '83370', '83383', '83435', '83600', '83738', '84', '84025'
, '84122', '84128', '84128custcar', '84199', '84484', '85', '850', '85023', '85069', '85222', '852
33', '8552', '85555', '86021', '861', '863', '864233', '86688', '86888', '87021', '87066', '87070'
, '87077', '87121', '87131', '8714714', '87239', '87575', '8800', '88039',
'88039skilgmetscs087147403231winawkage16', '88066', '88088', '88222', '8830', '88600', '88800', '8
883', '88877', '88877free', '88888', '89034', '89070', '89080', '89105', '89123', '89545', '89555'
, '89693', '89938', '8am', '8attract', '8ball', '8hr', '8lb', '8lovabl', '8neighbour', '8o', '8pm'
, '8th', '8wp', '9', '900', '9061100010', '9153', '924', '92h', '930', '945', '946', '95pax', '96'
, '97n7qp', '98321561', '9996', '9ae', '9am', '9am11pm', '9decent', '9funni', '9ja', '9pm', '9t',
'9th', '9yt', 'A', 'AD', 'AG', 'AH', 'AL', 'AM', 'AN', 'AS', 'AT', 'AV', 'Ab', 'Ah', 'Al', 'Am', 'A
n', 'As', 'At', 'Ay', 'B', 'B4', 'BE', 'BK', 'BT', 'BY', 'Bc', 'Be', 'Bt', 'Bx', 'By', 'C', 'CC', '
CD', 'CL', 'CM', 'CU', 'Co', 'Cs', 'D', 'DA', 'DD', 'DE', 'DO', 'Da', 'De', 'Do', 'Dr', 'E', 'ER',
'EY', 'Ee', 'Eh', 'Em', 'En', 'Er', 'Ew', 'F', 'FA', 'Fr', 'G', 'G2', 'GB', 'GO', 'Gd', 'Ge', 'Gn',
'Go', 'H', 'HI', 'HL', 'HM', 'HU', 'Ha', 'He', 'Hi', 'Hm', 'Ho', 'I', 'ID', 'IF', 'IL', 'IM', 'IN',
'IQ', 'IS', 'IT', 'Ic', 'Id', 'If', 'Im', 'In', 'Is', 'It', 'J', 'JD', 'K', 'KR', 'Ki', 'Ku', 'L',
'LE', 'Lk', 'M', 'M6', 'ME', 'MF', 'MO', 'MR', 'MY', 'Ma', 'Me', 'Mm', 'Mr', 'My', 'N', 'NO', 'No',
'Nt', 'Nw', 'O', 'O2', 'OF', 'OH', 'OK', 'ON', 'OR', 'Of', 'Oh', 'Oi', 'Ok', 'On', 'Or', 'Oz', 'P',
'PA', 'PC', 'PO', 'PS', 'Pa', 'Pg', 'Pl', 'Po', 'Q', 'R', 'RV', 'Re', 'Rs', 'S', 'S8', 'SD', 'SF',
'SI', 'SN', 'SO', 'SP', 'ST', 'Si', 'So', 'St', 'T', 'TA', 'TC', 'TH', 'TO', 'TV', 'TX', 'Ta', 'Tb'
, 'To', 'Ts', 'U', 'U4', 'UK', 'UP', 'UR', 'US', 'UU', 'Uh', 'Up', 'Ur', 'Us', 'V', 'VE', 'VU', 'W4
', 'WE', 'WK', 'Wa', 'We', 'Wk', 'Wn', 'X', 'X2', 'XX', 'Xx', 'Xy', 'Y', 'YA', 'YM', 'YO', 'Ya', 'Y
o', 'Z', 'a21', 'a30', 'aa', 'aah', 'aaniy', 'aaooooright', 'aathilov', 'aathiwher', 'abbey', 'abd
omen', 'abeg', 'abelu', 'aberdeen', 'abi', 'abil', 'abiola', 'abj', 'abl', 'abnorm', 'about', 'abo
ute', 'abroad', 'absenc', 'absolut', 'abstract', 'abt', 'abta', 'aburo', 'abus', 'ac', 'academ', '
```

```
uta', 'abroad', 'absenc', 'absolut', 'abstract', 'abt', 'abta', 'aburo', 'abus', 'ac', 'academ', '
acc', 'accent', 'accentur', 'accept', 'access', 'accid', 'accident', 'accommod',
'accommodationvouch', 'accomod', 'accordin', 'accordingli', 'accordinglyor', 'account', 'accumul',
'ach', 'achanammarakheshqatar', 'achiev', 'acid', 'acknowledg', 'acl03530150pm', 'acnt',
'acoentry41', 'across', 'acsmsreward', 'act', 'actin', 'action', 'activ', 'activ8', 'actor', 'actu
al', 'acwicmb3cktz8r74', 'ad', 'adam', 'add', 'addamsfa', 'addi', 'addict', 'address',
'addressul', 'adewal', 'adi', 'adjust', 'admin', 'administr', 'admir', 'admiss', 'admit', 'admiti'
, 'ador', 'adp', 'adress', 'adrian', 'adrink', 'adsens', 'adult', 'advanc', 'adventur', 'advic', '
advis', 'advisor', 'aeronaut', 'aeroplan', 'afew', 'affair', 'affect', 'affection',
'affectionsamp', 'affidavit', 'afford', 'afghanistan', 'afraid', 'africa', 'african', 'aft',
'after', 'afternon', 'afternoon', 'afterward', 'aftr', 'again', 'againcal', 'againlov', 'against',
'agalla', 'age', 'age16', 'age16150ppermesssubscript', 'age23', 'agenc', 'agent', 'agesr', 'agidha
n', 'ago', 'agocusoon', 'agre', 'agreen', 'ah', 'aha', 'ahead', 'ahge', 'ahhh', 'ahhhhjust',
'ahmad', 'ahnow', 'ahold', 'ahsen', 'ahth', 'ahwhat', 'aid', 'aig', 'aight', 'aint', 'air', 'air1'
, 'airport', 'airtel', 'aiya', 'aiyah', 'aiyar', 'aiyo', 'ajith', 'ak', 'aka', 'akonlon', 'al', 'a
laikkumprid', 'alaipayuth', 'albi', 'album', 'albumquit', 'alcohol', 'aldrin', 'alert',
'alertfrom', 'alett', 'alex', 'alfi', 'algarv', 'algebra', 'algorithm', 'ali', 'alian', 'alibi', '
aliv', 'alivebett', 'all', 'allah', 'allahmeet', 'allahrakhesh', 'allalo', 'allday', 'allo',
'allow', 'almost', 'alon', 'along', 'alot', 'alreadi', 'alreadysabarish', 'alright',
'alrightokay', 'alrit', 'alritehav', 'also', 'alsoor', 'alter', 'alternativehop', 'although', 'alw
a', 'alway', 'alwi', 'am', 'amanda', 'amaz', 'ambiti', 'ambrithmaduraimet', 'american', 'ami', 'am
igo', 'amk', 'ammaelif', 'ammo', 'amnow', 'among', 'amongst', 'amor', 'amount', 'amp', 'amplikat',
'amrca', 'amrita', 'amt', 'amus', 'amx', 'an', 'ana', 'anal', 'analysi', 'anand', 'and',
'anderson', 'andor', 'andr', 'andrewsboy', 'andro', 'angel', 'angri', 'ani', 'anim', 'anji', 'anjo
la', 'anna', 'anni', 'anniversari', 'annonc', 'announc', 'annoy', 'annoyin', 'anonym', 'anot', 'an
oth', 'ansr', 'answer', 'answerin', 'answr', 'antelop', 'anthoni', 'anti', 'antibiot', 'anybodi',
'anyhow', 'anymor', 'anyon', 'anyplac', 'anyth', 'anythi', 'anythin', 'anythingtomorrow',
'anytim', 'anyway', 'anywher', 'aom', 'apart', 'ape', 'apeshit', 'aphex', 'apnt', 'apo', 'apolog',
'apologet', 'apologis', 'app', 'appar', 'appeal', 'appear', 'appendix', 'appi', 'applebe',
'applespairsal', 'appli', 'applic', 'apply2', 'appoint', 'appreci', 'approach', 'appropri',
'approv', 'approx', 'appt', 'april', 'aproach', 'apt', 'aptitud', 'aquariu', 'ar', 'arab',
'arabian', 'arcad', 'archiv', 'ard', 'ardé', 'are', 'area', 'arent', 'arestaur', 'aretak', 'argent
ina', 'argh', 'argu', 'argument', 'ari', 'aris', 'arithmet', 'arm', 'armand', 'armenia', 'arng', '
arngd', 'arnt', 'around', 'aroundn', 'arpraveesh', 'arr', 'arrang', 'arrest', 'arriv', 'arrow', 'a
rsen', 'art', 'arti', 'artist', 'arul', 'arun', 'asa', 'asap', 'asapok', 'asda', 'ash', 'ashley',
'ashwini', 'asia', 'asian', 'ask', 'askd', 'askin', 'aslamalaikkuminsha', 'asleep', 'aspect', 'ass
', 'assess', 'asshol', 'assist', 'associ', 'assum', 'asther', 'asthma', 'astn', 'astoundingli', 'a
strolog', 'astronom', 'asu', 'asusual1', 'ate', 'athlet', 'athom', 'atlanta', 'atlast', 'atleast',
'atm', 'atroci', 'attach', 'attack', 'attempt', 'atten', 'attend', 'attent', 'attitud', 'attract',
'attractioni', 'attribut', 'atyour', 'auction', 'auctionpunj', 'audiit', 'audit', 'audrey',
'audri', 'august', 'aunt', 'aunti', 'aust', 'australia', 'authoris', 'auto', 'autocorrect', 'av',
'ava', 'avail', 'availa', 'availablei', 'availablethey', 'avalarr', 'avatar', 'avbl', 'ave',
'aveng', 'avent', 'avenu', 'avin', 'avo', 'avoid', 'await', 'awak', 'award', 'away', 'awesom', 'aw
kward', 'aww', 'awww', 'ax', 'axi', 'ayn', 'ayo', 'b', 'b4', 'b4190604', 'b4280703', 'b4u', 'ba',
'ba128nnfwfly150ppm', 'baaaaaaaab', 'baaaaab', 'babe', 'babeprobpop', 'babesozi', 'babi',
'babygoodby', 'babyhop', 'babyjontet', 'babysit', 'bac', 'back', 'backa', 'backdoor', 'backward',
'bad', 'badass', 'badli', 'badrith', 'bag', 'bagi', 'bahama', 'baig', 'bailiff', 'bak', 'bakra', '
bakrid', 'balanc', 'ball', 'baller', 'balloon', 'bam', 'bambl', 'ban', 'band', 'bandag', 'bang', '
bangb', 'bangbab', 'bani', 'bank', 'banneduk', 'banter', 'bao', 'bar', 'barbi', 'barcelona', 'bare
', 'bari', 'barkley', 'barm', 'barolla', 'barrel', 'barri', 'base', 'bash', 'basic', 'basket', 'ba
sketbal', 'basqihav', 'bat', 'batch', 'batchlor', 'bath', 'bathroom', 'batsman', 'batt',
'batteri', 'battl', 'bawl', 'bay', 'bb', 'bbc', 'bbdelux', 'bbdpooja', 'bbdtht', 'bblue', 'bbq', '
bc', 'bcaz', 'bck', 'bcm', 'bcm1896wc1n3xx', 'bcm4284', 'bcmsfwc1n3xx', 'bcoz', 'bcozi', 'bcum', '
bcz', 'bday', 'be', 'beach', 'bead', 'bear', 'beat', 'beauti', 'beautifulmay', 'bec', 'becau', 'be
caus', 'becausethey', 'becom', 'becoz', 'becz', 'bed', 'bedbut', 'bedreal', 'bedrm', 'bedrm900', '
bedroom', 'bedroomlov', 'beeen', 'beehoon', 'been', 'beendrop', 'beer', 'beerag', 'beerr',
'befor', 'beforehand', 'beforew', 'beg', 'beggar', 'begin', 'begun', 'behalf', 'behav', 'behind',
'bein', 'believ', 'beliv', 'bell', 'bellearli', 'belli', 'belliger', 'belong', 'belov', 'belovd',
'belt', 'ben', 'bend', 'beneath', 'beneficiari', 'benefit', 'benni', 'bergkamp', 'besid', 'best',
'best1', 'bestcongrat', 'bestrpli', 'bet', 'beta', 'beth', 'betta', 'better', 'bettersn',
'beverag', 'bevieswaz', 'bewar', 'beyond', 'bf', 'bff', 'bfore', 'bhaskar', 'bhayandar', 'bian', '
biatch', 'bid', 'big', 'bigger', 'biggest', 'bike', 'bill', 'billi', 'billion', 'bilo', 'bimbo', '
bin', 'biola', 'bird', 'birla', 'biro', 'birth', 'birthdat', 'birthday', 'bishan', 'bit', 'bitch',
'bite', 'bk', 'black', 'blackand', 'blackberri', 'blackim', 'blacko', 'blah', 'blake', 'blame', 'b
lank', 'blanket', 'blastin', 'bleak', 'bleh', 'bless', 'blessget', 'blimey', 'blind', 'block', 'bl
og', 'bloke', 'blond', 'bloo', 'blood', 'bloodblood', 'bloodi', 'bloodsend', 'bloomberg',
'bloombergcom', 'blow', 'blown', 'blu', 'blue', 'bluetooth', 'bluetoothhdset', 'blueu', 'bluff', '
blur', 'bluray', 'bmw', 'board', 'boat', 'boatin', 'bob', 'bodi', 'boggi', 'bognor', 'bold', 'bold
2', 'bollox', 'boltblu', 'bomb', 'bone', 'bong', 'bonu', 'boo', 'boob', 'book', 'bookedth',
'bookmark', 'bookshelf', 'boooo', 'boost', 'booti', 'bootydeli', 'borderlin', 'bore', 'borin', 'bo
rn', 'bornpleas', 'borrow', 'boss', 'boston', 'bot', 'both', 'bother', 'bottl', 'bottom',
'bought', 'boundari', 'bout', 'boutxx', 'bowa', 'bowl', 'box', 'box1146', 'box139', 'box177', 'box
245c2150pm', 'box326', 'box334', 'box334sk38ch', 'box385', 'box39822', 'box403', 'box420',
'box42wr29c', 'box434sk38wp150ppm18', 'box61m60', 'box95qu', 'box97n7qp', 'boy', 'boyf',
'boyfriend', 'boyi', 'boytoy', 'bpo', 'bra', 'brah', 'brain', 'braindanc', 'braini', 'brainless',
'brand', 'brandi', 'brat', 'brave', 'bray', 'brb', 'brdget', 'bread', 'breadstick', 'break', 'brea
ker', 'breakfast', 'breakin', 'breath', 'breathe1', 'breather', 'breez', 'breezi', 'bribe', 'bridg
```

```
', 'bridgwat', 'brief', 'bright', 'brighten', 'brilliant', 'brilliantithingi', 'brilliantli', 'bri
n', 'bring', 'brisk', 'brison', 'bristol', 'british', 'britney', 'bro', 'broad', 'broadband', 'bro
ke', 'broken', 'brolli', 'broth', 'brotha', 'brother', 'brought', 'browni', 'brows', 'browser', 'b
rowsin', 'bruce', 'brum', 'bruv', 'bslvyl', 'bsn', 'bsnl', 'bstfrnd', 'bt', 'bthere', 'bthmm', 'bt
nation', 'btnationalr', 'btooth', 'btw', 'btwn', 'bu', 'buck', 'bud', 'buddi', 'budget', 'buen', '
buff', 'buffet', 'buffi', 'bugi', 'build', 'built', 'bulb', 'bull', 'bullshit', 'bun', 'bunch', 'b
undl', 'bunker', 'burden', 'burger', 'burgundi', 'burial', 'burn', 'burnt', 'burrito',
'bus822656166382', 'buse', 'busetop', 'busi', 'busti', 'busyi', 'but', 'butt', 'butther',
'button', 'buy', 'buyer', 'buz', 'buzi', 'buzz', 'buzzzz', 'bw', 'bx420', 'bx420ip45w', 'bx526', '
byatch', 'bye', 'c', 'c52', 'cab', 'cabin', 'cabl', 'cafe', 'cage', 'cake', 'caken', 'cal', 'calcu
l', 'cali', 'calicut', 'california', 'call', 'call09050000327', 'call2optout4qf2',
'call2optout674', 'call2optoutf4q', 'call2optouthf8', 'call2optoutj', 'call2optoutj5q',
'call2optoutlf56', 'call2optoutn9dx', 'call2optoutyhl', 'callback', 'callcost', 'callcoz',
'calld', 'calldrov', 'caller', 'callertun', 'callfreefon', 'callin', 'callingforgot', 'callon', 'c
alls150ppm', 'callsmessagesmiss', 'callurg', 'calm', 'cam', 'camcord', 'came', 'camera',
'cameravideo', 'camp', 'campu', 'camri', 'can', 'canada', 'canal', 'canari', 'cancel', 'cancer', '
candont', 'canlov', 'cannam', 'cannot', 'cannt', 'cant', 'cantdo', 'canteen', 'cap', 'capac', 'cap
it', 'cappuccino', 'captain', 'car', 'card', 'cardiff', 'cardin', 'care', 'careabout', 'career', '
careinsha', 'careless', 'carent', 'careswt', 'careumma', 'carewhoev', 'carli', 'carlin', 'carlo',
'carlosl', 'carolin', 'carolina', 'carpark', 'carri', 'carryin', 'carso', 'carton', 'cartoon', 'ca
se', 'cash', 'cashbal', 'cashbincouk', 'cashin', 'cashto', 'cast', 'castor', 'casualti', 'cat', 'c
atch', 'categori', 'caught', 'caus', 'cave', 'caveboy', 'cbe', 'cc100pmin', 'ccna', 'cd', 'cdgt',
'cedar', 'ceil', 'celeb', 'celeb4', 'celebr', 'cell', 'censu', 'center', 'centr', 'centuri', 'cer'
, 'cereal', 'ceri', 'certainli', 'certif', 'cha', 'chachi', 'chad', 'chain', 'challeng', 'champ',
'champlaxig', 'champney', 'chanc', 'chang', 'channel', 'chap', 'chapel', 'chapter', 'charact', 'ch
arg', 'charged150pmsg2', 'chariti', 'charl', 'charli', 'charm', 'chart', 'chase', 'chastiti', 'cha
t', 'chat80155', 'chatim', 'chatlin', 'chatter', 'cheap', 'cheaper', 'cheat', 'chechi', 'check', '
checkbox', 'checkin', 'checkmat', 'checkup', 'cheek', 'cheer', 'cheeri', 'chees', 'cheesi', 'cheet
o', 'chef', 'chennai', 'chennaibecaus', 'chennaii', 'chequ', 'cherish', 'cherthalain', 'chess', 'c
hest', 'chex', 'cheyyamoand', 'chez', 'chg', 'chic', 'chick', 'chicken', 'chief', 'chik',
'chikku', 'chikkuali', 'chikkub', 'chikkudb', 'chikkugo', 'chikkuil', 'chikkuk', 'chikkusimpl', 'c
hikkuwat', 'child', 'childish', 'childporn', 'children', 'chile', 'chill', 'chillaxin', 'chillin',
'china', 'chinatown', 'chinchilla', 'chines', 'chinki', 'chiong', 'chip', 'chitchat', 'chk',
'chloe', 'chocol', 'choic', 'choos', 'chop', 'chord', 'chore', 'chosen', 'chrgd50p', 'christ', 'ch
ristian', 'christma', 'christmasmerri', 'christmassi', 'chuck', 'chuckin', 'church', 'ciao', 'cin'
, 'cine', 'cinema', 'citi', 'citizen', 'citylink', 'cla', 'claim', 'claimcod', 'clair', 'clarif',
'clarifi', 'clash', 'class', 'classic', 'classmat', 'claypot', 'cld', 'clean', 'clear', 'clearer',
'clearli', 'clever', 'click', 'cliff', 'clip', 'clock', 'clos1', 'close', 'closebi',
'closedinclud', 'closer', 'closingdate040902', 'cloth', 'cloud', 'clover', 'club', 'club4', 'club4
mobilescom', 'clue', 'cm', 'cme', 'cmon', 'cn', 'cnl', 'cnn', 'co', 'coach', 'coast', 'coat', 'coa
x', 'cocacola', 'coccoon', 'cochin', 'cock', 'cocksuck', 'coco', 'code', 'code4xx26', 'coffe', 'co
her', 'coimbator', 'coin', 'coincid', 'colani', 'cold', 'coldheard', 'colin', 'collag', 'collaps',
'colleagu', 'collect', 'colleg', 'collegexx', 'color', 'colour', 'colourredtextcolourtxtstar', 'co
m', 'comb', 'combin', 'come', 'comedi', 'comedyc', 'comei', 'cometil', 'comfey', 'comfort',
'comin', 'comingdown', 'comingtmorow', 'command', 'comment', 'commerci', 'commit', 'common',
'commun', 'comp', 'compani', 'companion', 'compar', 'compass', 'compens', 'competit', 'complac', '
complain', 'complaint', 'complementari', 'complet', 'complex', 'compliment', 'complimentari',
'compofstuff', 'comprehens', 'compromis', 'compulsori', 'comput', 'computerless', 'comuk220cm2',
'con', 'conact', 'concentr', 'concern', 'concert', 'conclus', 'condit', 'conditionand', 'conduct',
'conect', 'confer', 'confid', 'configur', 'confirm', 'confirmd', 'confirmdeni', 'conform',
'confus', 'congrat', 'congratul', 'connect', 'consensu', 'consent', 'conserv', 'consid',
'consist', 'consol', 'constant', 'constantli', 'contact', 'contain', 'content', 'contin',
'continu', 'contract', 'contribut', 'control', 'conveni', 'convers', 'convert', 'convey',
'convinc', 'convincingjust', 'cook', 'cooki', 'cool', 'coolmob', 'coop', 'cooper', 'cop', 'cope',
'copi', 'corect', 'cornwal', 'corpor', 'corrct', 'correct', 'correctionor', 'correctli',
'corrupt', 'corvett', 'cosign', 'cost', 'costa', 'costum', 'couch', 'cougarpen', 'cough', 'could',
'coulda', 'couldnt', 'count', 'countin', 'countinlot', 'countri', 'coupl', 'coupla', 'courag', 'co
urs', 'court', 'courtroom', 'cousin', 'cover', 'coveragd', 'coz', 'cozi', 'cozsomtim', 'cp', 'cr',
'cr01327bt', 'cr9', 'crab', 'crack', 'craigslist', 'cram', 'cramp', 'crap', 'crash', 'crave', 'cra
zi', 'craziest', 'crazyin', 'cream', 'creat', 'creativ', 'cred', 'credit', 'creep', 'creepi', 'cre
subi', 'cri', 'cribb', 'cricket', 'crickit', 'crisi', 'crisisspk', 'cro1327', 'crore', 'cross', 'c
rowd', 'croydon', 'crucial', 'crucifi', 'cruis', 'cruisin', 'crush', 'cs', 'csh11', 'cst',
'cstore', 'ctagg', 'ctargg', 'cthen', 'ctla', 'cttargg', 'ctter', 'cttergg', 'cuck', 'cud', 'cuddl
', 'cudnt', 'culdnt', 'cultur', 'cum', 'cumin', 'cup', 'cupboard', 'cuppa', 'curfew', 'curiou', 'c
urrent', 'curri', 'curtsey', 'cust', 'custcar', 'custcare08718720201', 'custom', 'customercar', 'c
ustomersqueriesnetvisionukcom', 'cut', 'cute', 'cutefrnd', 'cutest', 'cuti', 'cutter', 'cuz', 'cw2
5wx', 'cya', 'cyclist', 'cyst', 'da', 'daal', 'daalway', 'dabbl', 'dabook', 'dad', 'daddi',
'dado', 'dagood', 'dahe', 'dahow', 'dai', 'daili', 'dajst', 'dammit', 'damn', 'dan', 'danalla', 'd
anc', 'dancc', 'dancin', 'dane', 'dang', 'danger', 'dao', 'dapleas', 'dare', 'dark', 'darker', 'da
rkest', 'darl', 'darlin', 'darlinim', 'darren', 'dartboard', 'dasara', 'dat', 'data', 'date', 'dat
ebox1282essexcm61xn', 'datingi', 'datoday', 'datz', 'daurgent', 'dave', 'dawhat', 'dawher',
'dawn', 'day', 'day2', 'day2find', 'dayexcept', 'dayha', 'daysh', 'daysso', 'dayswil', 'daysèn', '
daytim', 'dayu', 'daywith', 'de', 'dead', 'deadwel', 'deal', 'dealer', 'dealfarm', 'deam', 'dear',
'dear1', 'dearer', 'deari', 'dearli', 'dearlov', 'dearm', 'dearrakhesh', 'dearregret', 'dearshal',
'dearslp', 'deartak', 'death', 'debat', 'dec', 'decad', 'decemb', 'decid', 'decim', 'decis', 'deck
', 'declar', 'decor', 'dedic', 'deduct', 'deep', 'deepak', 'deepest', 'deer', 'deeraj', 'def', 'de
feat', 'defer', 'definit', 'definitli', 'defo', 'degre', 'dehydr', 'del', 'delay', 'delet', 'delhi
```

```
', 'delici', 'deliv', 'deliveredtomorrow', 'deliveri', 'deltomorrow', 'delux', 'dem', 'demand', 'd
en', 'dena', 'dengra', 'deni', 'dent', 'dental', 'dentist', 'depart', 'depend', 'deposit',
'depress', 'dept', 'der', 'derek', 'derp', 'describ', 'descript', 'desert', 'deserv', 'design', 'd
esir', 'desk', 'despar', 'desper', 'despit', 'dessert', 'destin', 'destini', 'detail', 'detailsi',
'determin', 'detroit', 'deu', 'develop', 'devic', 'devil', 'devour', 'dey', 'deyhop', 'deyi', 'dha
', 'dhina', 'dhoni', 'dhort', 'di', 'dial', 'diall', 'dialogu', 'diamond', 'diaper', 'dice', 'dick
', 'dict', 'dictionari', 'did', 'diddi', 'didn', 'didnt', 'didntgiv', 'didt', 'die', 'diesel', 'di
et', 'diff', 'differ', 'differb', 'difficult', 'difficulti', 'dificult', 'digi', 'digit',
'digniti', 'dileepthank', 'dime', 'dimens', 'din', 'dine', 'dinero', 'ding', 'dinner',
'dinnermsg', 'dino', 'dint', 'dip', 'dippeditinadew', 'direct', 'directli', 'director', 'dirt', 'd
irti', 'dirtiest', 'disagre', 'disappear', 'disappoint', 'disast', 'disastr', 'disc', 'disclos',
'disconnect', 'discount', 'discreet', 'discuss', 'diseas', 'diskyou', 'dislik', 'dismay',
'dismissi', 'display', 'distanc', 'distract', 'disturb', 'disturbancemight', 'ditto', 'divert', 'd
ivis', 'divorc', 'diwali', 'dizzamn', 'dizze', 'dl', 'dled', 'dlf', 'dload', 'dnt', 'do', 'dob', '
dobbi', 'doc', 'dock', 'doctor', 'document', 'dodda', 'dodgey', 'doe', 'doesdiscountshitinnit', 'd
oesnt', 'dog', 'dogbreath', 'dogg', 'doggi', 'doggin', 'dogwood', 'doin', 'doinat', 'doinghow', 'd
oingwhat', 'doinnearli', 'dointerest', 'doke', 'dokey', 'doll', 'dollar', 'dolld', 'dom',
'domain', 'don', 'donat', 'done', 'donew', 'donno', 'dont', 'dont4get2text', 'dontcha',
'dontignor', 'dontpleas', 'donyt', 'doom', 'door', 'dorm', 'dormitori', 'dorothykiefercom',
'dose', 'dosometh', 'dot', 'doubl', 'doublefaggot', 'doublemin', 'doubletxt', 'doubt', 'doug', 'do
ugh', 'down', 'download', 'downon', 'downstem', 'dozen', 'dp', 'dr', 'dracula', 'drama',
'dramastorm', 'dramat', 'drastic', 'draw', 'drawpleas', 'dread', 'dream', 'dreamlov',
'dreamsmuah', 'dreamstak', 'dreamsu', 'dreamz', 'dress', 'dresser', 'dri', 'drink', 'drinkin', 'dr
inkpa', 'drive', 'driver', 'drivin', 'drizzl', 'drm', 'drmstake', 'drop', 'drove', 'drpd', 'drug',
'drugdeal', 'drum', 'drunk', 'drunkard', 'drunken', 'drvgsto', 'dryer', 'dsnt', 'dt', 'dual',
'dub', 'dubsack', 'duchess', 'duck', 'dude', 'dudett', 'due', 'duffer', 'dull', 'dumb', 'dump',
'dun', 'dungere', 'dunno', 'duo', 'durban', 'durham', 'dusk', 'dust', 'duvet', 'dvd', 'dvg', 'dwn'
, 'dysentri', 'e', 'e14', 'each', 'eachoth', 'ear', 'earli', 'earlier', 'earlierw', 'earliest', 'e
arn', 'earth', 'earthsofa', 'easi', 'easier', 'easiest', 'easili', 'east', 'eastend', 'easter', 'e
at', 'eaten', 'eatin', 'ebay', 'ec2a', 'echo', 'eckankar', 'ecstaci', 'ecstasi', 'edg', 'edha', 'e
dison', 'edit', 'edrunk', 'educ', 'edukkukaye', 'edward', 'ee', 'eek', 'eeri', 'eerulli',
'effect', 'effici', 'efreefon', 'eg', 'eg23f', 'eg23g', 'egbon', 'egg', 'eggpotato', 'eggspert', '
ego', 'eh', 'eh74rr', 'eight', 'eighth', 'eightish', 'eir', 'either', 'el', 'ela', 'elabor', 'elai
n', 'elama', 'elaya', 'eldest', 'elect', 'electr', 'eleph', 'eleven', 'elliot', 'ello', 'els', 'el
sewher', 'elvi', 'em', 'email', 'embarass', 'embarrass', 'embassi', 'emerg', 'emigr', 'emili', 'em
ot', 'employ', 'employe', 'empti', 'en', 'enam', 'enc', 'end', 'endless', 'endof', 'endow', 'enemi
', 'energi', 'eng', 'engag', 'engalnd', 'engin', 'england', 'english', 'enjoy', 'enjoyin',
'enketa', 'enna', 'ennal', 'enough', 'enter', 'entertain', 'entey', 'entir', 'entitl',
'entrepreneur', 'entri', 'entrop', 'enufcredeit', 'enuff', 'envelop', 'envi', 'epi', 'epsilon', 'e
qual', 'ericson', 'ericsson', 'erm', 'erot', 'err', 'error', 'ertini', 'eruku', 'erupt',
'erutupalam', 'eryth', 'esaplanad', 'escal', 'escap', 'ese', 'eshxxxxxxxxxxx', 'especi', 'espel',
'esplanad', 'essay', 'essenti', 'establish', 'eta', 'etc', 'etern', 'ethnic', 'ethreat', 'ettan',
'euro', 'euro2004', 'eurodisinc', 'europ', 'evalu', 'evapor', 'eve', 'eveb', 'evei', 'even', 'even
t', 'eventu', 'ever', 'everi', 'every1', 'everybodi', 'everyboy', 'everyday', 'everyon',
'everyso', 'everyth', 'everythin', 'everytim', 'everywher', 'evey', 'evict', 'evil', 'evn',
'evng', 'evo', 'evon', 'evr', 'evrey', 'evri', 'evry1', 'evrydi', 'ex', 'exact', 'exactli', 'exam'
, 'excel', 'except', 'exchang', 'excit', 'excus', 'exe', 'execut', 'exercis', 'exet', 'exhaust', '
exhibit', 'exist', 'exmpel', 'exorc', 'exorcist', 'exp', 'expect', 'expens', 'experi',
'experiencehttpwwwvouch4mecometlpdiningasp', 'expert', 'expir', 'expiredso', 'expiri', 'explain',
'explicit', 'explicitli', 'explos', 'expos', 'express', 'ext', 'extermin', 'extra', 'extract', 'ex
trem', 'exwif', 'ey', 'eye', 'eyeddont', 'f', 'fab', 'faber', 'face', 'faceassssholeee',
'facebook', 'facil', 'fact', 'factori', 'fade', 'faggi', 'faglord', 'fail', 'failur', 'faint', 'fa
ir', 'faith', 'faitheven', 'fake', 'fakemi', 'fakey', 'fal', 'falconerf', 'fall', 'fallen', 'famam
u', 'famili', 'familiar', 'familymay', 'famou', 'fan', 'fanci', 'fantasi', 'fantast', 'far', 'farm
', 'farrel', 'fart', 'fassyol', 'fast', 'faster', 'fastest', 'fastpl', 'fat', 'fate', 'father', 'f
athima', 'fatti', 'fault', 'faultal', 'faultf', 'fav', 'fave', 'favor', 'favorit', 'favour', 'favo
urit', 'fb', 'fear', 'featheri', 'featur', 'feb', 'febapril', 'februari', 'fedex', 'fee', 'feed',
'feel', 'feelin', 'feelingood', 'feelingwav', 'feet', 'fell', 'fellow', 'felt', 'femal', 'feng', '
festiv', 'fetch', 'fever', 'fffff', 'ffffffffff', 'ffffuuuuuuu', 'fgkslpo', 'fgkslpopw', 'fidalf',
'field', 'fieldof', 'fiendmak', 'fifa', 'fifteen', 'fifth', 'fifti', 'fight', 'fightng', 'figur',
'file', 'fill', 'film', 'filth', 'filthi', 'filthyguy', 'final', 'finalis', 'financ', 'financi', '
find', 'fine', 'fineabsolutli', 'fineinshah', 'finest', 'finewhen', 'finger', 'finish', 'finishd',
'fink', 'finn', 'fire', 'firefox', 'fireplac', 'firesar', 'firmwar', 'firsg', 'first', 'fish', 'fi
shhead', 'fishrman', 'fit', 'fite', 'five', 'fix', 'fixd', 'fixedlin', 'fizz', 'flag', 'flake', 'f
laki', 'flame', 'flash', 'flat', 'flatter', 'flavour', 'flea', 'fletcher', 'flew', 'fli',
'flight', 'flim', 'flip', 'flippin', 'flirt', 'float', 'flood', 'floor', 'floppi', 'florida', 'flo
w', 'flower', 'fluid', 'flung', 'flurri', 'flute', 'flyim', 'flyng', 'fml', 'fmyou', 'fne', 'fo',
'fold', 'foley', 'folk', 'follow', 'followin', 'fond', 'fondli', 'fone', 'fonin', 'food', 'fool',
'foot', 'footbal', 'footblcrckt', 'footi', 'footprint', 'for', 'forc', 'foreg', 'foreign',
'forev', 'forevr', 'forfeit', 'forget', 'forgiv', 'forgiven', 'forgot', 'forgotten', 'forgt', 'for
m', 'formal', 'formallypl', 'format', 'formclark', 'formsdon', 'forth', 'fortun', 'forum',
'forward', 'found', 'foundurself', 'four', 'fourth', 'foward', 'fowler', 'fox', 'fp', 'fr',
'fraction', 'fran', 'frankgood', 'franki', 'franxx', 'franyxxxxx', 'fraud', 'freak', 'freaki', 'fr
edericksburg', 'free', 'free2day', 'freedom', 'freeentri', 'freefon', 'freek', 'freeli',
'freemessag', 'freemsg', 'freemsgfav', 'freemsgfeelin', 'freenokia', 'freephon', 'freerington', 'f
reeringtonerepli', 'freesend', 'freez', 'freind', 'fren', 'french', 'frequent', 'fresh',
'fresher', 'fret', 'fri', 'friday', 'fridayhop', 'fridg', 'friend', 'friendofafriend',
'friendsar', 'friendship', 'friendshipmotherfatherteacherschildren', 'fring', 'frm', 'frmcloud', '
frnd', 'frndship', 'frndshp', 'frndsship', 'frndz', 'frnt', 'fro', 'frog', 'frogaxel', 'from', 'fr
omm', 'fromwrk', 'front', 'frontiervil', 'frosti', 'fruit', 'frwd', 'ft', 'fuck', 'fuckin',
'fuckinniceselfishdeviousbitchanywayi', 'fudg', 'fuell', 'fujitsu', 'ful', 'fulfil', 'full',
'fullonsmscom', 'fumbl', 'fun', 'function', 'fund', 'fundament', 'funer', 'funk', 'funki', 'funni'
, 'furnitur', 'fusion', 'futur', 'fuuuuck', 'fwiw', 'fyi', 'g', 'g696ga', 'ga', 'gail', 'gailxx',
'gain', 'gal', 'galcan', 'galileo', 'galno', 'galsu', 'gam', 'game', 'gamestar', 'gandhipuram', 'g
anesh', 'gang', 'gap', 'garag', 'garbag', 'garden', 'gari', 'garment', 'gastroenter', 'gate', 'gau
g', 'gautham', 'gave', 'gay', 'gayd', 'gayl', 'gaytextbuddycom', 'gaze', 'gbp', 'gbp150week',
'gbp450week', 'gbp5month', 'gbpsm', 'gbpweek', 'gd', 'gdeve', 'gdnow', 'gdthe', 'ge', 'gee', 'geee
e', 'geeeee', 'geelat', 'gei', 'gek1510', 'gender', 'gene', 'gener', 'geniu', 'gent', 'gentl', 'ge
ntleman', 'gentli', 'genu', 'genuin', 'geoenvironment', 'georg', 'gep', 'ger', 'germani', 'get', '
get4an18th', 'gete', 'geti', 'getsleep', 'getstop', 'gettin', 'getzedcouk', 'gf', 'ghodbandar',
'ghost', 'gibb', 'gibe', 'gift', 'giggl', 'gigolo', 'gimm', 'gimmi', 'gin', 'girl', 'girld', 'girl
frnd', 'girli', 'gist', 'giv', 'give', 'given', 'givit', 'glad', 'gland', 'glasgow', 'glass', 'glo
', 'global', 'glori', 'gloriou', 'gloucesterroad', 'gmgngegn', 'gmgngegnt', 'gmw', 'gnarl', 'go',
'go2', 'go2sri', 'goa', 'goal', 'goalsteam', 'gobi', 'god', 'godi', 'godnot', 'godtaken',
'godyou', 'goe', 'goggl', 'goigng', 'goin', 'goin2b', 'gokila', 'gold', 'golddigg', 'golden', 'gol
dvik', 'golf', 'gon', 'gona', 'gone', 'goneu', 'gong', 'gonna', 'gonnamissu', 'good',
'gooddhanush', 'goodenviron', 'goodeven', 'goodfin', 'goodfriend', 'goodi', 'goodmat', 'goodmorn',
'goodmorningmi', 'goodnight', 'goodnit', 'goodno', 'goodnoon', 'goodo', 'goodtimeoli', 'goodwhen',
'googl', 'gopalettan', 'gorgeou', 'gosh', 'gossip', 'gossx', 'got', 'gota', 'gotani', 'gotmarri',
'goto', 'gotta', 'gotten', 'gotto', 'gover', 'govtinstituit', 'gowait', 'gower', 'gpr', 'gpu', 'gr
8', 'gr8fun', 'gr8prize', 'grab', 'grace', 'graduat', 'grahmbel', 'gram', 'gran', 'grand',
'grandfath', 'grandma', 'granit', 'grant', 'graphic', 'grasp', 'grate', 'grave', 'gravel', 'gravi'
, 'graviti', 'gray', 'graze', 'gre', 'great', 'greatbhaji', 'greatby', 'greatest', 'greatli',
'greec', 'green', 'greeni', 'greet', 'grief', 'grin', 'grinder', 'grinul', 'grl', 'grocer', 'groov
', 'groovi', 'ground', 'groundamla', 'group', 'grow', 'grown', 'grownup', 'growrandom', 'grr', 'gr
umbl', 'grumpi', 'gs', 'gsex', 'gsoh', 'gt', 'gua', 'guai', 'guarante', 'gucci', 'gud', 'gudk', 'g
udni8', 'gudnit', 'gudnitetcpractic', 'gudnyt', 'guess', 'guessin', 'guid', 'guidanc', 'guild', 'g
uilti', 'guitar', 'gumbi', 'guoyang', 'gurl', 'gut', 'guy', 'gv', 'gving', 'gwr', 'gym',
'gymnast', 'gyna', 'gyno', 'h', 'ha', 'habbahw', 'habit', 'hack', 'had', 'hadnt', 'hadya', 'haf',
'haha', 'hahahaus', 'hahatak', 'hai', 'hail', 'hair', 'haircut', 'hairdress', 'haiyoh', 'haiz', 'h
alf', 'half8th', 'hall', 'halla', 'hallaq', 'halloween', 'ham', 'hamper', 'hamster', 'hand',
'handl', 'handset', 'handsom', 'hang', 'hanger', 'hangin', 'hank', 'hannaford', 'hanumanji',
'happen', 'happend', 'happenin', 'happi', 'happier', 'happiest', 'happili', 'hard', 'hardcor', 'ha
rder', 'hardest', 'hardli', 'hari', 'harish', 'harlem', 'harri', 'hasbroin', 'hasnt', 'hassl', 'ha
t', 'hate', 'haughaighgtujhyguj', 'haul', 'haunt', 'hav', 'hav2hear', 'hava', 'havbeen', 'have', '
havebeen', 'havent', 'haventcn', 'havin', 'havnt', 'hcl', 'hdd', 'he', 'head', 'headach',
'headin', 'headset', 'headstart', 'heal', 'healer', 'healthi', 'heap', 'hear', 'heard', 'hearin',
'heart', 'heartgn', 'heartheart', 'heartsnot', 'heat', 'heater', 'heaven', 'heavi', 'heavili',
'hectic', 'hee', 'heehe', 'hehe', 'height', 'held', 'helen', 'hell', 'hella', 'hello',
'hellodrivby0quit', 'hellogorg', 'hellohow', 'helloooo', 'helloy', 'help', 'help08700469649',
'help08700621170150p', 'help08712400602450p', 'help08714742804', 'help08718728876', 'helplin',
'heltiniiyo', 'hen', 'henc', 'henri', 'hep', 'her', 'here', 'herepl', 'hereremember', 'herethanksi',
'heri', 'herlov', 'hermi', 'hero', 'heroi', 'heron', 'hersh', 'herwho', 'herwil', 'hesit', 'hex',
'hey', 'heygreat', 'hgsuite3422land', 'hgsuite3422landsroww1j6hl', 'hhahhaahahah', 'hi', 'hict',
'hidden', 'hide', 'hidid', 'high', 'highest', 'hii', 'hilariousalso', 'hill', 'hillsborough', 'him
', 'himso', 'himthen', 'hint', 'hip', 'hiphop', 'hire', 'hisher', 'histori', 'hit', 'hitechn', 'hi
tler', 'hitman', 'hitteranyway', 'hittng', 'hiwhat', 'hiya', 'hlday', 'hlp', 'hme', 'hmm',
'hmmbad', 'hmmm', 'hmmmbut', 'hmmmhow', 'hmmmi', 'hmmmkbut', 'hmmmm', 'hmmmstill', 'hmph', 'hmv',
'hmv1', 'ho', 'hockey', 'hogidhechinnu', 'hogli', 'hogolo', 'hol', 'holbi', 'hold', 'holder', 'hol
e', 'holi', 'holiday', 'holidayso', 'holla', 'hollalat', 'home', 'homebut', 'homecheck',
'homeleft', 'homelov', 'homeown', 'homewot', 'hon', 'honest', 'honesti', 'honestli', 'honey',
'honeybe', 'honeydid', 'honeymoon', 'honi', 'hont', 'hoo', 'hooch', 'hoodi', 'hook', 'hoop', 'hop'
, 'hope', 'hopeafternoon', 'hopeso', 'hopeu', 'hor', 'horni', 'horniest', 'horo', 'horribl', 'hors
', 'hospit', 'hostbas', 'hostel', 'hostil', 'hot', 'hotel', 'hotmix', 'hottest', 'hour',
'hourish', 'hous', 'housemaid', 'housew', 'housework', 'how', 'howard', 'howda', 'howdi', 'howev',
'howr', 'howu', 'howv', 'howz', 'hp', 'hp20', 'hppnss', 'hr', 'hrishi', 'hsbc', 'html',
'httpalto18coukwavewaveaspo44345', 'httpcareer', 'httpdoit', 'httpgotbabescouk', 'httpimg',
'httptm', 'httpwap', 'httpwwwbubbletextcom', 'httpwwwetlpcoukexpressoff', 'httpwwwetlpcoukreward',
'httpwwwgr8prizescom', 'httpwwwurawinnercom', 'httpwwwwtlpcouktext', 'huai', 'hubbi', 'hudgi', 'hu
g', 'huge', 'hugh', 'huh', 'hui', 'huim', 'hum', 'human', 'hun', 'hundr', 'hundredh', 'hungov', 'h
ungri', 'hunk', 'hunlov', 'hunni', 'hunnyhop', 'hunnyjust', 'hunnywot', 'hunonbu', 'hunt',
'hurri', 'hurrican', 'hurt', 'husband', 'hussey', 'hustl', 'hut', 'hv', 'hvae', 'hw', 'hwd',
'hwkeep', 'hyde', 'hypertens', 'hypotheticalhuagauahahuagahyuhagga', 'iZ', 'ia', 'iam', 'ibh', 'ib
hltd', 'ibiza', 'ibm', 'ibn', 'ibor', 'ibuprofen', 'ic', 'iccha', 'ice', 'icic', 'icicibankcom', '
icki', 'icon', 'id', 'idc', 'idconvey', 'idea', 'ideal', 'identif', 'identifi', 'idiot', 'idk', 'i
dp', 'idu', 'ie', 'iff', 'ifink', 'ifwhenhow', 'ig11', 'ignor', 'ijust', 'ikea', 'ikno', 'iknow',
'il', 'ileav', 'ill', 'illspeak', 'ilol', 'im', 'ima', 'imag', 'imagin', 'imaginationmi', 'imat',
'imf', 'imin', 'imma', 'immedi', 'immunis', 'imp', 'impati', 'implic', 'import', 'importantli', 'i
mpos', 'imposs', 'impost', 'impress', 'improv', 'imprtant', 'in2', 'inc', 'inch', 'incid',
'inclu', 'includ', 'inclus', 'incomm', 'inconsider', 'inconveni', 'incorrect', 'increas',
'incred', 'increment', 'ind', 'inde', 'independ', 'india', 'indian', 'indianpl', 'indic', 'individ
u', 'individualtim', 'indyarockscom', 'inev', 'infact', 'infect', 'infern', 'influx', 'info', 'inf
oringtonekingcouk', 'inform', 'informedrgdsrakheshkerala', 'infotxt82228couk', 'infovipclub4u', 'i
nfowww100percentrealcom', 'infra', 'infront', 'ing', 'ingredi', 'initi', 'ink', 'inlud', 'inmind',
'inner', 'inning', 'innoc', 'innu', 'inour', 'inperialmus', 'inperson', 'inr', 'insect', 'insha',
'inshah', 'insid', 'inspect', 'inst', 'instal', 'instant', 'instantli', 'instead', 'instruct', 'in
sur', 'intellig', 'intend', 'intent', 'interest', 'interflora', 'interfu', 'intern', 'internet', '
internetservic', 'interview', 'interviw', 'intha', 'intim', 'into', 'intrepid', 'intro', 'intrud',
'invad', 'invent', 'invest', 'investig', 'invit', 'invnt', 'invoic', 'involv', 'iouri', 'ip', 'ip4
', 'ipad', 'ipaditan', 'iphon', 'ipod', 'iraq', 'ireneer', 'iriv', 'iron', 'irrit', 'irulina', 'is
aiahd', 'isar', 'iscom', 'ish', 'ishtamayoohappi', 'island', 'islov', 'isnt', 'issu', 'isvimport',
'it', 'italian', 'itboth', 'itc', 'itcould', 'item', 'iter', 'ithi', 'ithink', 'iti', 'itjust', 'i
tleav', 'itlet', 'itll', 'itmail', 'itmay', 'itna', 'itnow', 'itor', 'itplspl', 'itried2tel', 'its
not', 'ittb', 'itu', 'itwhichturnedinto', 'itxt', 'itxx', 'itz', 'ivatt', 'ive', 'iwana',
'iwasmarinethat', 'iz', 'izzit', 'j', 'j89', 'jabo', 'jack', 'jacket', 'jackpot', 'jackson',
'jacuzzi', 'jada', 'jade', 'jaklin', 'jam', 'jame', 'jamster', 'jamstercouk', 'jamsterget',
'jamz', 'jan', 'janarig', 'jane', 'janinexx', 'januari', 'janx', 'jap', 'japanes', 'jason',
'java', 'jay', 'jaya', 'jaykwon', 'jaz', 'jazz', 'jb', 'je', 'jealou', 'jean', 'jeetey',
'jeevithathil', 'jelli', 'jen', 'jenn', 'jenni', 'jenxxx', 'jeremiah', 'jeri', 'jerk', 'jerri', 'j
ersey', 'jess', 'jesu', 'jet', 'jetton', 'jewelri', 'jez', 'ji', 'jia', 'jiayin', 'jide', 'jiu', '
jjc', 'jo', 'joanna', 'job', 'jobyet', 'jock', 'jod', 'jog', 'john', 'join', 'joinedhop',
'joinedso', 'joke', 'joker', 'jokethet', 'jokin', 'jolli', 'jolt', 'jon', 'jone', 'jontin', 'jorda
n', 'jordantxt', 'jorgeshock', 'jot', 'journey', 'joy', 'jp', 'js', 'jsco', 'jst', 'jstfrnd',
'jsut', 'ju', 'juan', 'judgementali', 'juici', 'jule', 'juli', 'juliana', 'julianaland', 'jump', '
jumper', 'june', 'jungl', 'junna', 'jurong', 'just', 'justbeen', 'justifi', 'justthought',
'juswok', 'juz', 'k', 'k52', 'k61', 'k718', 'kaaj', 'kadeem', 'kafter', 'kaiez', 'kaila', 'kaitlyn
', 'kalaachutaarama', 'kalainar', 'kalisidar', 'kall', 'kalli', 'kalstiyathen', 'kama', 'kanagu',
'kane', 'kanji', 'kano', 'kanoanyway', 'kanoil', 'kanowhr', 'kappa', 'karaok', 'karnan', 'karo', '
kate', 'katexxx', 'kath', 'kavalan', 'kay', 'kaypoh', 'kb', 'kbut', 'kdo', 'ke', 'keen', 'keep',
'keepintouch', 'kegger', 'keluviri', 'ken', 'keng', 'kent', 'kept', 'kerala', 'keralacircl',
'keri', 'kettoda', 'key', 'keypad', 'keyword', 'kfc', 'kg', 'kgive', 'kgood', 'khelat', 'ki', 'kic
chu', 'kick', 'kickbox', 'kickoff', 'kid', 'kidz', 'kill', 'kilo', 'kim', 'kind', 'kinda',
'kindli', 'king', 'kingdom', 'kintu', 'kiosk', 'kip', 'kisi', 'kiss', 'kit', 'kitti', 'kittum', 'k
kadvanc', 'kkani', 'kkapo', 'kkare', 'kkcongratul', 'kkfrom', 'kkgoodstudi', 'kkhow', 'kkim', 'kki
t', 'kkthi', 'kkwhat', 'kkwhen', 'kkwhere', 'kkwhi', 'kkyesterday', 'kl341', 'knacker', 'knee', 'k
new', 'knicker', 'knock', 'know', 'knowh', 'known', 'knowneway', 'knowthi', 'knowwait',
'knowyetund', 'knw', 'ko', 'kochi', 'kodstini', 'kodthini', 'konw', 'korch', 'korean', 'korli', 'k
ort', 'kote', 'kothi', 'ksri', 'kthen', 'ktv', 'kuch', 'kudiyarasu', 'kusruthi', 'kvb', 'kwish', '
kyou', 'kz', 'l8', 'l8er', 'l8r', 'l8tr', 'la', 'la1', 'la3', 'la32wu', 'lab', 'labor', 'lac', 'la
ck', 'lacsthat', 'lacsther', 'laden', 'ladi', 'ladiesu', 'lag', 'lage', 'lager', 'laid',
'laidwant', 'lakh', 'lambda', 'lambu', 'lamp', 'lancast', 'land', 'landlin', 'landlineonli',
'landmark', 'lane', 'langport', 'languag', 'lanka', 'lanr', 'lap', 'lapdanc', 'laptop', 'lar', 'la
ra', 'lareadi', 'larg', 'largest', 'lark', 'lasagna', 'last', 'lastest', 'late', 'latebut',
'latei', 'latelyxxx', 'later', 'lateso', 'latest', 'latr', 'laugh', 'laundri', 'lauri', 'lautech',
'lavend', 'law', 'laxinorf', 'lay', 'layin', 'lazi', 'lccltd', 'ldn', 'ldnw15h', 'le', 'lead', 'le
adership', 'leafcutt', 'leafdayno', 'leagu', 'leannewhat', 'learn', 'least', 'least5tim',
'leastwhich', 'leav', 'lect', 'lectur', 'left', 'leftov', 'leg', 'legal', 'legitimat', 'leh', 'leh
haha', 'lei', 'lekdog', 'lemm', 'length', 'lennon', 'leo', 'leona', 'leonardo', 'less', 'lesser',
'lesson', 'let', 'letter', 'leu', 'level', 'li', 'liao', 'liaoso', 'liaotoo', 'lib', 'libertin', '
librari', 'lick', 'lido', 'lie', 'life', 'lifeand', 'lifebook', 'lifei', 'lifethi', 'lifetim', 'li
fey', 'lifpartnr', 'lift', 'light', 'lighter', 'lightli', 'lik', 'like', 'likeyour', 'likingb', 'l
il', 'lili', 'lim', 'limit', 'limp', 'lindsay', 'line', 'linear', 'linerent', 'liney', 'lingeri',
'lingo', 'link', 'linux', 'lion', 'lionm', 'lionp', 'lip', 'lipo', 'liquor', 'list', 'listen', 'li
stening2th', 'listn', 'lit', 'liter', 'litr', 'littl', 'live', 'liver', 'liverpool', 'lk', 'lkpobo
x177hp51fl', 'llspeak', 'lm', 'lmao', 'lmaonic', 'lnli', 'lo', 'load', 'loan', 'lobbi', 'local', '
locat', 'locaxx', 'lock', 'lodg', 'log', 'login', 'logo', 'logoff', 'logon', 'logop',
'logosmusicnew', 'loko', 'lol', 'lolnic', 'lololo', 'londn', 'london', 'lone', 'loneli', 'long', '
longer', 'lonlin', 'loo', 'look', 'lookatm', 'lookin', 'lool', 'loooooool', 'looovvv', 'loos',
'loosu', 'lor', 'lord', 'lorgoin', 'lorw', 'lose', 'loser', 'loss', 'lost', 'lot', 'loti', 'lotr',
'lotsli', 'lotsof', 'lotta', 'lotto', 'lotwil', 'lotz', 'lou', 'loud', 'loung', 'lousi', 'lov', 'l
ovabl', 'love', 'loveabl', 'lovejen', 'lovem', 'lover', 'loverakhesh', 'loverboy', 'lovin', 'lovin
gli', 'lovli', 'low', 'lowcost', 'lower', 'loxahatche', 'loyal', 'loyalti', 'lrg', 'ls1',
'ls15hb', 'ls278bb', 'lst', 'lt', 'lt3', 'ltd', 'ltdecimalgt', 'ltdhelpdesk', 'ltemailgt', 'ltgt',
'lttimegt', 'lttr', 'lturlgt', 'lubli', 'luci', 'luck', 'luck2', 'lucki', 'luckili', 'lucozad', 'l
ucozadecoukwrc', 'lucyxx', 'luk', 'lul', 'lunch', 'lunchtim', 'lunchyou', 'lunsford', 'lush', 'lut
on', 'luv', 'luvd', 'luvnight', 'lux', 'luxuri', 'lv', 'lvblefrnd', 'lyf', 'lyfu', 'lyk', 'lyric',
'lyricalladie21f', 'm100', 'm221bp', 'm227xi', 'm26', 'm263uz', 'm39m51', 'm8', 'm95', 'ma',
'maaaan', 'maangalyam', 'maat', 'mac', 'macedonia', 'macha', 'machan', 'machiani', 'machin', 'mach
o', 'mack', 'macleran', 'mad', 'mad1', 'mad2', 'madam', 'madamregret', 'made', 'madodu', 'madok',
'madstini', 'madthen', 'mag', 'maga', 'magazin', 'maggi', 'magic', 'magicalsongsblogspotcom', 'mah
', 'mahal', 'mahfuuzmean', 'mail', 'mailbox', 'maili', 'main', 'maintain', 'major', 'make', 'maki'
, 'makin', 'malaria', 'malarki', 'male', 'mall', 'mallika', 'man', 'manag', 'manchest', 'manda', '
mandan', 'mandara', 'mandi', 'maneesha', 'maneg', 'mango', 'mani', 'maniac', 'manki', 'manual', 'm
ap', 'mapquest', 'maraikara', 'marandratha', 'march', 'maretar', 'margaret', 'margin', 'mari', 'ma
rk', 'market', 'marley', 'marrgeremembr', 'marri', 'marriag', 'marriageprogram', 'marsm',
'marvel', 'mask', 'massag', 'massagetiepo', 'massiv', 'master', 'masteriast', 'mat', 'match', 'mat
e', 'math', 'mathemat', 'mathew', 'matra', 'matric', 'matrix3', 'matter', 'mattermsg', 'matthew',
'matur', 'max', 'max10min', 'max6month', 'maxim', 'maximum', 'may', 'mayb', 'mb', 'mc', 'mca', 'mc
at', 'mcr', 'meal', 'mean', 'meaning', 'meaningless', 'meant', 'meanwhil', 'mear', 'measur', 'meat
', 'meatbal', 'mecaus', 'med', 'medic', 'medicin', 'medont', 'mee', 'meet', 'meetgreet', 'meetin',
'meetitz', 'mega', 'meh', 'mei', 'meim', 'meiv', 'mel', 'melik', 'mell', 'melnit', 'melodi', 'melt
```

```
', 'member', 'membership', 'membershiptak', 'memor', 'memori', 'men', 'mene', 'mental', 'mention',
'mentionedtomorrow', 'mentor', 'menu', 'meok', 'meow', 'meowd', 'mere', 'merememberin', 'meremov',
'merri', 'mesag', 'mesh', 'meso', 'mess', 'messag', 'messageit', 'messageno', 'messagepandi',
'messagesim', 'messagesom', 'messagestext', 'messagethank', 'messeng', 'messi', 'met', 'method', '
meummifyingby', 'mfl', 'mg', 'mi', 'mia', 'michael', 'mid', 'middl', 'midnight', 'might',
'miiiiiiisssssssss', 'mila', 'mile', 'mileag', 'milk', 'milkdayno', 'miller', 'million',
'miltazindgi', 'min', 'mina', 'minapn', 'mind', 'mindi', 'mindsetbeliev', 'mine', 'mineal',
'minecraft', 'mini', 'minimum', 'minnaminungint', 'minor', 'mins100txtmth', 'minstand',
'minstext', 'mint', 'minu', 'minut', 'miracl', 'mirror', 'misbehav', 'mise', 'miser', 'misfit', 'm
isplac', 'miss', 'misscal', 'missi', 'missin', 'mission', 'missionari', 'misss', 'misstak',
'missunderstd', 'mist', 'mistak', 'mistakeu', 'misundrstud', 'mite', 'mitsak', 'mittelschmertz', '
miwa', 'mix', 'mj', 'mjzgroup', 'mk17', 'mk45', 'ml', 'mmm', 'mmmm', 'mmmmm', 'mmmmmm', 'mmmmmmm',
'mmsto', 'mn', 'mnth', 'mo', 'moan', 'mob', 'mobcudb', 'mobi', 'mobil', 'mobilesdirect',
'mobilesvari', 'mobileupd8', 'mobno', 'mobsicom', 'mobstorequiz10ppm', 'mode', 'model',
'modelsoni', 'modl', 'modul', 'mofo', 'moji', 'mojibiola', 'mokka', 'molestedsomeon', 'mom', 'mome
nt', 'mon', 'monday', 'mondaynxt', 'moneeppolum', 'money', 'moneya', 'moneyi', 'monkeespeopl', 'mo
nkey', 'monkeyaround', 'monl8rsx', 'mono', 'monoc', 'monster', 'month', 'monthli',
'monthlysubscription50pmsg', 'monthnot', 'mood', 'moon', 'moral', 'moraldont', 'moralon', 'more',
'morn', 'mornin', 'morningtak', 'morphin', 'morrow', 'moseley', 'most', 'mostli', 'mother',
'motherfuck', 'motherinlaw', 'motiv', 'motor', 'motorola', 'mountain', 'mous', 'mouth', 'move', 'm
ovi', 'moviewat', 'moyep', 'mp3', 'mquiz', 'mr', 'mre', 'mrng', 'mrt', 'mrur', 'ms', 'msg', 'msg15
0p', 'msging', 'msgrcvd18', 'msgs150p', 'msgsd', 'msgsometext', 'msgsubscript',
'msgticketkioskvalid', 'msgwe', 'msn', 'mssuman', 'mt', 'mtalk', 'mth', 'mtnl', 'mu', 'much', 'muc
hand', 'muchi', 'muchimped', 'muchxxlov', 'mudyadhu', 'mufti', 'muhommad', 'muht', 'multi', 'multi
media', 'multipli', 'mum', 'mumbai', 'mumha', 'mummi', 'mumtaz', 'mundh', 'munster', 'murali', 'mu
rder', 'mush', 'mushi', 'music', 'must', 'musta', 'musthu', 'mustprovid', 'mutai', 'mutat', 'muz',
'mw', 'mwah', 'my', 'mycallsu', 'mylif', 'mymobi', 'mypar', 'myself', 'myspac', 'mysteri',
'mytonecomenjoy', 'n', 'n8', 'na', 'naal', 'nacho', 'nag', 'nagar', 'nah', 'nahi', 'nail', 'nake',
'nalla', 'nalli', 'name', 'name1', 'name2', 'namemi', 'nammanna', 'nan', 'nang', 'nanni', 'nap', '
narcot', 'nasdaq', 'naseeb', 'nasti', 'nat', 'natali', 'natalja', 'nation', 'nationwid', 'nattil',
'natuit', 'natur', 'natwest', 'naughti', 'nauseou', 'nav', 'navig', 'nb', 'nbme', 'nd', 'ne',
'near', 'nearbi', 'nearer', 'nearli', 'neces', 'necess', 'necessari', 'necessarili', 'neck', 'neck
lac', 'ned', 'need', 'needa', 'neededsalari', 'needi', 'needl', 'neekunna', 'neft', 'neg', 'neglec
t', 'neglet', 'neighbor', 'neither', 'nelson', 'neo69', 'nervou', 'neshanthtel', 'net',
'netcollex', 'netflix', 'neth', 'netno', 'network', 'neva', 'nevamindw', 'never', 'nevil', 'nevr',
'new', 'neway', 'newest', 'newport', 'newquaysend', 'news', 'newsbi', 'newscast', 'newshyp',
'newspap', 'next', 'ngage', 'nh', 'ni8', 'ni8swt', 'nic', 'nice', 'nicenicehow', 'nichol', 'nick',
'nickey', 'nicki', 'nig', 'nigeria', 'nigh', 'night', 'nighter', 'nightnight', 'nightnobodi', 'nig
htsexcel', 'nightsw', 'nightswt', 'nigpun', 'nigro', 'nike', 'nikiyu4net', 'nimbomson', 'nimya', '
nimyapl', 'ninish', 'nino', 'nipost', 'nit', 'nite', 'nite2', 'nitro', 'nitw', 'nitz', 'njan', 'nm
de', 'no', 'no1', 'no165', 'no434', 'no440', 'no762', 'no81151', 'no83355', 'no910', 'nob', 'nobl'
, 'nobodi', 'nobut', 'noe', 'nofew', 'nohe', 'noi', 'noic', 'nois', 'noisi', 'noit', 'nojst', 'nok
', 'nokia', 'nokia150p', 'nokia6600', 'nokia6650', 'nolin', 'nolistened2th', 'non', 'noncomitt', '
none', 'nonenowher', 'nonetheless', 'nookii', 'noon', 'nooooooo', 'noooooooo', 'nope', 'nora',
'norcorp', 'nordstrom', 'norm', 'norm150pton', 'normal', 'north', 'northampton', 'nose', 'nosh', '
nosi', 'not', 'note', 'notebook', 'noth', 'nothi', 'nothin', 'notic', 'notif', 'notifi',
'notixiqu', 'nottel', 'nottingham', 'notxtcouk', 'noun', 'novelti', 'novemb', 'now', 'now1',
'now4t', 'nowaday', 'nowadayslot', 'nowcan', 'nowi', 'nownyt', 'nowonion', 'noworriesloanscom', 'n
owrepli', 'nowsavamobmemb', 'nowsend', 'nowski', 'nowstil', 'nowtc', 'nowus', 'nr31', 'nri', 'nt',
'nte', 'ntswt', 'ntt', 'ntwk', 'nu', 'nuclear', 'nudist', 'nuerologist', 'num', 'number',
'numberpl', 'numberrespect', 'numberso', 'nurs', 'nurseri', 'nurungu', 'nusstu', 'nuther',
'nutter', 'nver', 'nvm', 'nvq', 'nw', 'nxt', 'ny', 'nyc', 'nydc', 'nyt', 'nytec2a3lpmsg150p', 'nyt
ho', 'nyusa', 'nz', 'nìte', 'o2coukgam', 'o2fwd', 'oath', 'obedi', 'obes', 'obey', 'object', 'obli
sing', 'oblivi', 'obvious', 'occas', 'occupi', 'occur', 'oceand', 'oclock', 'octob', 'odalebeku',
'odi', 'ofcours', 'off', 'offc', 'offcampu', 'offdam', 'offens', 'offer', 'offerth', 'offic', 'off
icestil', 'officethenampet', 'officeunderstand', 'officewhat', 'offici', 'offlin', 'ofic',
'oficegot', 'ofsi', 'often', 'oga', 'ogunrind', 'oh', 'oha', 'ohi', 'oic', 'oil', 'oja', 'ok', 'ok
ay', 'okcom', 'okday', 'okden', 'okey', 'oki', 'okmail', 'okok', 'okor', 'oktak', 'okthenwhat', 'o
kvarunnathu', 'ola', 'olag', 'olav', 'olayiwola', 'old', 'ollubut', 'olol', 'olowoyey', 'olymp', '
omg', 'omw', 'onam', 'onc', 'oncal', 'ondu', 'one', 'onedg', 'oneta', 'oni', 'onionr', 'onit', 'on
li', 'onlin', 'onlinewhi', 'onluy', 'only1mor', 'onlybettr', 'onlydon', 'onlyfound', 'onto', 'onum
', 'onward', 'onword', 'ooh', 'oooh', 'oooooh', 'ooooooh', 'oop', 'open', 'openin', 'oper',
'opinion', 'opp', 'opponent', 'opportun', 'opportunityal', 'opportunitypl', 'oppos', 'opposit', 'o
pt', 'optimist', 'optin', 'option', 'optout', 'or', 'or2optouthv9d', 'or2stoptxt', 'oral',
'orang', 'orangei', 'orc', 'orchard', 'order', 'ore', 'oredi', 'oreo', 'organ', 'organis', 'orh',
'orig', 'origin', 'orno', 'ortxt', 'oru', 'os', 'oscar', 'oso', 'otbox', 'other', 'otherwis', 'oth
r', 'otsid', 'ou', 'ouch', 'our', 'ourback', 'oursso', 'out', 'outag', 'outbid', 'outdoor',
'outfit', 'outfor', 'outgo', 'outhav', 'outif', 'outl8rjust', 'outrag', 'outreach', 'outsid', 'out
somewher', 'outstand', 'outta', 'ovarian', 'over', 'overa', 'overdid', 'overdos',
'overemphasiseor', 'overh', 'overtim', 'ovr', 'ovul', 'ovulatewhen', 'ow', 'owe', 'owl', 'own', 'o
wnyouv', 'owo', 'oxygen', 'oyea', 'oyster', 'oz', 'p', 'pa', 'pace', 'pack', 'packag', 'packalso',
'padhegm', 'page', 'pai', 'paid', 'pain', 'painhop', 'painit', 'paint', 'pale', 'palm', 'pan', 'pa
nalambut', 'panason', 'pandi', 'panic', 'panick', 'panren', 'pansi', 'pant', 'panther', 'panti', '
pap', 'papa', 'paper', 'paperwork', 'paracetamol', 'parachut', 'parad', 'paragon', 'paragraph', 'p
aranoid', 'parantella', 'parchi', 'parco', 'parent', 'parentnot', 'parentsi', 'pari', 'parisfre',
'parish', 'park', 'park6ph', 'parkin', 'part', 'parti', 'particip', 'particular', 'particularli',
'partner', 'partnership', 'paru', 'pase', 'pass', 'passabl', 'passion', 'passport', 'passthey', 'p
```

```
assword', 'passwordsatmsm', 'past', 'pataistha', 'patent', 'path', 'pathaya', 'patient',
'patrick', 'pattern', 'patti', 'paul', 'paus', 'pay', 'payasam', 'payback', 'paye', 'payed2day', '
payment', 'payoh', 'paypal', 'pc', 'pdatenow', 'peac', 'peach', 'peak', 'pear', 'pee', 'peep', 'pe
hl', 'pei', 'pen', 'penc', 'pend', 'pendent', 'pendingi', 'peni', 'penni', 'peopl', 'per', 'percen
t', 'percentag', 'perf', 'perfect', 'perform', 'perfum', 'perhap', 'peril', 'period', 'peripher',
'perman', 'permiss', 'perpetu', 'persev', 'persian', 'person', 'person2di', 'personmeet',
'perspect', 'perumbavoor', 'peski', 'pest', 'pete', 'petei', 'petexxx', 'petey', 'peteynoi', 'petr
ol', 'petrolr', 'pg', 'ph', 'ph08700435505150p', 'ph08704050406', 'pharmaci', 'phase', 'phd', 'phe
w', 'phil', 'philosoph', 'philosophi', 'phne', 'phoenix', 'phone', 'phone750', 'phonebook',
'phoni', 'photo', 'photoshop', 'php', 'phrase', 'physic', 'piah', 'pic', 'pick', 'pickl',
'picsfree1', 'pictur', 'pictxt', 'pie', 'piec', 'pierr', 'pig', 'piggi', 'pilat', 'pile', 'pillow'
, 'pimpl', 'pimpleseven', 'pin', 'pink', 'pinku', 'pint', 'pisc', 'piss', 'piti', 'pix', 'pixel',
'pizza', 'pl', 'place', 'placement', 'placeno', 'plaid', 'plan', 'plane', 'planet', 'planeti', 'pl
anettalkinstantcom', 'plate', 'platt', 'play', 'player', 'playerwhi', 'playi', 'playin', 'playng',
'plaza', 'pleas', 'pleasant', 'pleassssssseeeee', 'pleasur', 'plenti', 'plm', 'plough', 'plsi', 'p
lu', 'plum', 'plumber', 'plumbingremix', 'plural', 'plyr', 'plz', 'pm', 'pmt', 'po', 'po19',
'pobox', 'pobox1', 'pobox11414tcrw1', 'pobox12n146tf15', 'pobox12n146tf150p', 'pobox202',
'pobox334', 'pobox36504w45wq', 'pobox365o4w45wq', 'pobox45w2tg150p', 'pobox75ldns7', 'pobox84',
'poboxox36504w45wq', 'pocay', 'poci', 'pock', 'pocket', 'pocketbabecouk', 'pod', 'poem', 'poet', '
point', 'poke', 'poker', 'pokkiri', 'pole', 'poli', 'polic', 'politician', 'polo', 'poly200p', 'po
ly3', 'polyc', 'polyh', 'polyph', 'polyphon', 'polytruepixringtonesgam', 'pongal', 'pongaldo', 'po
nnungal', 'poo', 'pooki', 'pool', 'poop', 'poor', 'poorli', 'poortiyagi', 'pop', 'popcorn',
'popcornjust', 'porn', 'porridg', 'port', 'portal', 'porteg', 'portion', 'pose', 'posh', 'posibl',
'posit', 'possess', 'possibl', 'possiblehop', 'post', 'postal', 'postcard', 'postcod', 'posterod',
'postpon', 'potato', 'potenti', 'potter', 'pouch', 'pound', 'pour', 'pout', 'power',
'poyyarikaturkolathupalayamunjalur', 'ppl', 'pple', 'pple700', 'ppm', 'ppm150', 'ppt150x3normal',
'prabha', 'prabhaim', 'prabu', 'pract', 'practic', 'practicum', 'practis', 'prais', 'prakasam', 'p
rakasamanu', 'prakesh', 'prap', 'prasad', 'prasanth', 'prashanthettan', 'pray', 'prayer',
'prayingwil', 'prayr', 'pre', 'prebook', 'predict', 'prefer', 'prem', 'premaricakindli',
'premier', 'premium', 'prepaid', 'prepar', 'prepay', 'prepon', 'preschoolcoordin', 'prescrib', 'pr
escripiton', 'prescript', 'presenc', 'present', 'presid', 'presley', 'presnt', 'press', 'pressi',
'pressur', 'prestig', 'pretend', 'pretsorginta', 'pretsovru', 'pretti', 'prevent', 'preview',
'previou', 'previous', 'prey', 'price', 'priceso', 'pride', 'priest', 'prin', 'princ', 'princegn',
'princess', 'print', 'printer', 'prior', 'prioriti', 'priscilla', 'privaci', 'privat', 'prix', 'pr
iya', 'prize', 'prizeawait', 'prizeswith', 'prizeto', 'pro', 'prob', 'probabl', 'problem',
'problemat', 'problembut', 'problemfre', 'problemi', 'problm', 'problum', 'probthat', 'process', '
processexcel', 'processit', 'processnetwork', 'prod', 'product', 'prof', 'profession',
'professor', 'profil', 'profit', 'program', 'progress', 'project', 'prolli', 'prometazin',
'promin', 'promis', 'promo', 'promot', 'prompt', 'promptli', 'prone', 'proof', 'proov', 'prop', 'p
roper', 'properli', 'properti', 'propos', 'propsd', 'prospect', 'protect', 'prove', 'proverb', 'pr
ovid', 'provinc', 'proze', 'prsn', 'ps3', 'pshewmiss', 'psp', 'psxtra', 'psychiatrist', 'psychic',
'psychologist', 'pt2', 'ptbo', 'pthi', 'pub', 'pubcaf', 'public', 'publish', 'pudunga', 'pull', 'p
ump', 'punch', 'punish', 'punto', 'puppi', 'pura', 'purchas', 'pure', 'puriti', 'purpleu',
'purpos', 'purs', 'push', 'pushbutton', 'pussi', 'put', 'puttin', 'puzzel', 'puzzl', 'px3748', 'qa
tar', 'qatarrakhesh', 'qbank', 'qet', 'qi', 'qing', 'qlynnbv', 'qualiti', 'quarter', 'que',
'queen', 'queri', 'question', 'questionstd', 'quick', 'quickli', 'quiet', 'quit', 'quiteamuz', 'qu
iz', 'quizclub', 'quizwin', 'quizz', 'quot', 'r', 'r836', 'ra', 'racal', 'race', 'radiat',
'radio', 'rael', 'raglan', 'rahul', 'raiden', 'railway', 'rain', 'rais', 'raj', 'raja', 'rajini',
'rajipl', 'rajitha', 'rajnik', 'rakhesh', 'raksha', 'ralli', 'ralph', 'ramen', 'ran', 'randi', 'ra
ndom', 'randomli', 'randomlli', 'rang', 'ranjith', 'ranju', 'rape', 'rat', 'rate', 'ratetc',
'rather', 'ratio', 'raviyog', 'rawr', 'ray', 'rayan', 'rayman', 'rcbbattl', 'rcd', 'rct', 'rcv', '
rcvd', 'rd', 'rdi', 'reach', 'react', 'reaction', 'read', 'reader', 'readi', 'readyal', 'real', 'r
eal1', 'reali', 'realis', 'realiti', 'realiz', 'realli', 'reallyne', 'reappli', 'rearrang',
'reason', 'reassur', 'rebel', 'reboot', 'rebtel', 'rec', 'recd', 'recdthirtyeight', 'receipt', 're
ceiv', 'receivea', 'recent', 'recept', 'recess', 'recharg', 'rechargerakhesh', 'reciev', 'reckon',
'recognis', 'record', 'recount', 'recoveri', 'recpt', 'recreat', 'recycl', 'red', 'redeem',
'redim', 'redr', 'reduc', 'ree', 'ref', 'ref9280114', 'ref9307622', 'refer', 'referin', 'reffer',
'refil', 'reflect', 'reflex', 'reformat', 'refresh', 'refund', 'refundedthi', 'refus', 'reg', 'reg
ard', 'regist', 'registr', 'regret', 'regular', 'reject', 'rel', 'relat', 'relationshipit',
'relax', 'releas', 'reliant', 'reliev', 'religi', 'reloc', 'reltnship', 'rem', 'remain', 'remb', '
rememb', 'rememberi', 'remembr', 'remet', 'remind', 'remov', 'rencontr', 'renew', 'rent',
'rental', 'rentl', 'repair', 'repeat', 'repent', 'replac', 'repli', 'replyb', 'replys150', 'report
', 'reppurcuss', 'repres', 'republ', 'request', 'requir', 'reschedul', 'research', 'resend',
'resent', 'reserv', 'reset', 'resid', 'resiz', 'reslov', 'resolut', 'resolv', 'resort', 'respect',
'responcewhat', 'respond', 'respons', 'rest', 'restaur', 'restock', 'restrict', 'restuwud',
'restwish', 'resub', 'resubmit', 'result', 'resum', 'retard', 'retir', 'retriev', 'return', 'reuni
on', 'reveal', 'revers', 'review', 'revis', 'reward', 'rg21', 'rgd', 'rgent', 'rhode', 'rhythm', '
rice', 'rich', 'riddanc', 'ridden', 'ride', 'right', 'rightio', 'rightli', 'riley', 'rimac',
'ring', 'ringsreturn', 'rington', 'ringtonefrom', 'ringtoneget', 'ringtonek', 'rinu', 'rip',
'rise', 'risk', 'rite', 'ritten', 'river', 'ro', 'road', 'roadsrvx', 'roast', 'rob', 'robinson', '
rock', 'rodds1', 'rodger', 'rofl', 'roger', 'role', 'roll', 'roller', 'romant', 'romcapspam',
'ron', 'room', 'roomat', 'roommat', 'rose', 'rough', 'round', 'rounderso', 'rout', 'row',
'roww1j6hl', 'roww1jhl', 'royal', 'rp176781', 'rpl', 'rpli', 'rr', 'rreveal', 'rs', 'rs5', 'rsi',
'rstm', 'rtking', 'rtm', 'rto', 'ru', 'rub', 'rubber', 'rude', 'rudi', 'rugbi', 'ruin', 'rule', 'r
um', 'rumbl', 'rummer', 'rumour', 'run', 'runninglet', 'rupaul', 'rush', 'ryan', 'ryder', 's3xi',
's89', 'sac', 'sachin', 'sachinjust', 'sack', 'sacrific', 'sad', 'sae', 'saeed', 'safe', 'safeti',
'sagamu', 'saibaba', 'said', 'saidif', 'sake', 'salad', 'salam', 'salari', 'sale', 'salesman', 'sa
```

```
lespe', 'sall', 'salmon', 'salon', 'salt', 'sam', 'samachara', 'samantha', 'sambarlif', 'same', 's
ameso', 'samu', 'sandiago', 'sane', 'sang', 'sankranti', 'santa', 'santha', 'sao', 'sapna', 'sar',
'sara', 'sarasota', 'sarcasm', 'sarcast', 'sari', 'saristar', 'sariyag', 'sashimi', 'sat',
'satan', 'sathi', 'sathya', 'satisfi', 'satjust', 'satlov', 'satsgettin', 'satsound', 'satthen', '
saturday', 'satü', 'sauci', 'sausagelov', 'savamob', 'save', 'saw', 'say', 'sayask', 'sayhey', 'sa
yi', 'sayin', 'sbut', 'sc', 'scalli', 'scammer', 'scarcasim', 'scare', 'scari', 'scenario', 'scene
ri', 'sch', 'schedul', 'school', 'scienc', 'scold', 'scool', 'scorabl', 'score', 'scotch', 'scotla
nd', 'scotsman', 'scous', 'scrape', 'scrappi', 'scratch', 'scream', 'screen', 'screwd', 'scroung',
'scrumptiou', 'sculptur', 'sd', 'sday', 'sdryb8i', 'se', 'sea', 'search', 'season', 'seat', 'sec',
'second', 'secondari', 'secret', 'secretari', 'secretli', 'section', 'secur', 'sed', 'see', 'seed'
, 'seek', 'seeker', 'seem', 'seen', 'seeno', 'sef', 'seh', 'sehwag', 'select', 'self',
'selfindepend', 'selfish', 'selfless', 'sell', 'sem', 'semest', 'semi', 'semiobscur', 'sen',
'send', 'sender', 'sendernam', 'senor', 'senrddnot', 'sens', 'sensesrespect', 'sensibl', 'sensit',
'sent', 'sentdat', 'sentenc', 'senthil', 'senthilhsbc', 'seperated鈥', 'sept', 'septemb', 'serena',
'seri', 'seriou', 'serious', 'serv', 'server', 'servic', 'set', 'settl', 'seven', 'seventeen', 'se
ver', 'sex', 'sexi', 'sexiest', 'sextextukcom', 'sexual', 'sexychat', 'sez', 'sfine', 'sfirst', 's
from', 'sh', 'sha', 'shade', 'shadow', 'shag', 'shah', 'shahjahan', 'shakara', 'shake',
'shakespear', 'shall', 'shame', 'shampain', 'shangela', 'shanghai', 'shanilrakhesh', 'shant', 'sha
pe', 'share', 'shatter', 'shave', 'shb', 'shd', 'she', 'sheet', 'sheffield', 'shelf', 'shell', 'sh
elv', 'sherawat', 'shesil', 'shexi', 'shhhhh', 'shi', 'shifad', 'shija', 'shijutta', 'shinco', 'sh
indig', 'shine', 'shini', 'ship', 'shirt', 'shit', 'shite', 'shitin', 'shitjustfound', 'shitload',
'shitstorm', 'shivratri', 'shja', 'shld', 'shldxxxx', 'shock', 'shoe', 'shola', 'shoot', 'shop', '
shoppin', 'shopth', 'shopw', 'shoranur', 'shore', 'shoreth', 'short', 'shortag', 'shortcod', 'shor
ter', 'shortli', 'shot', 'shoul', 'should', 'shoulder', 'shouldnt', 'shout', 'shove', 'show', 'sho
wer', 'showr', 'showroomsc', 'shracomorsglsuplt10', 'shrek', 'shrink', 'shrub', 'shu', 'shud', 'sh
udvetold', 'shuhui', 'shun', 'shut', 'si', 'sian', 'sib', 'sic', 'sick', 'sicomo', 'side', 'sif',
'sigh', 'sight', 'sign', 'signal', 'signific', 'signin', 'siguviri', 'silenc', 'silent', 'silli',
'silver', 'sim', 'simonwatson5120', 'simpl', 'simpler', 'simpli', 'simpson', 'simul', 'sinc', 'sin
co', 'sindu', 'sing', 'singapor', 'singl', 'sink', 'sip', 'sipix', 'sir', 'siri', 'sirjii', 'sirsa
lam', 'sister', 'sit', 'site', 'sitll', 'sitter', 'sittin', 'situat', 'siva', 'sivatat', 'six', 's
ize', 'sk3', 'sk38xh', 'skalli', 'skateboard', 'ski', 'skilgm', 'skill', 'skillgam',
'skillgame1winaweek', 'skin', 'skinni', 'skint', 'skip', 'skirt', 'sky', 'skye', 'skype', 'skyve',
'slaaaaav', 'slack', 'slap', 'slave', 'sleep', 'sleepi', 'sleepin', 'sleepingand', 'sleepingwith',
'sleepsweet', 'sleepwellamptak', 'slept', 'slice', 'slide', 'slightli', 'slip', 'slipper',
'slipperi', 'slo', 'slo4msg', 'slob', 'slot', 'slove', 'slow', 'slower', 'slowli', 'slurp', 'sm',
'smack', 'small', 'smaller', 'smart', 'smartcal', 'smarter', 'smartthough', 'smash', 'smear', 'sme
ll', 'smeon', 'smidgin', 'smile', 'smiley', 'smith', 'smithswitch', 'smoke', 'smokin', 'smoothli',
'sms08718727870', 'smsd', 'smsing', 'smsservic', 'smsshsexnetun', 'smth', 'sn', 'snake', 'snap', '
snappi', 'snatch', 'snd', 'sneham', 'snicker', 'sno', 'snog', 'snoringthey', 'snow', 'snowbal', 's
nowboard', 'snowman', 'snuggl', 'so', 'soani', 'soc', 'socht', 'social', 'sofa', 'soft',
'softwar', 'soil', 'soire', 'sol', 'soladha', 'sold', 'solihul', 'solv', 'some', 'some1',
'somebodi', 'someday', 'someon', 'someonethat', 'someonon', 'someplac', 'somerset', 'someth', 'som
ethin', 'sometim', 'sometimerakheshvisitor', 'sometm', 'somewhat', 'somewher', 'somewheresomeon',
'somewhr', 'somon', 'somtim', 'sonathaya', 'sonetim', 'song', 'soni', 'sonot', 'sonyericsson', 'so
o', 'soon', 'soonc', 'sooner', 'soonlot', 'soonxxx', 'sooo', 'soooo', 'sooooo', 'sopha', 'sore', '
sori', 'sorri', 'sorrow', 'sorrowsi', 'sorryi', 'sorryin', 'sort', 'sorta', 'sortedbut',
'sorydarealyfrm', 'soso', 'soul', 'sound', 'soundtrack', 'soup', 'sourc', 'south', 'southern', 'so
uveni', 'soz', 'space', 'spacebuck', 'spageddi', 'spain', 'spam', 'spanish', 'spare', 'spark', 'sp
arkl', 'spatula', 'speak', 'spec', 'special', 'specialcal', 'specialis', 'specif', 'specifi',
'speechless', 'speed', 'speedchat', 'spele', 'spell', 'spend', 'spent', 'spi', 'spice', 'spider',
'spiderman', 'spif', 'spile', 'spin', 'spinout', 'spiral', 'spirit', 'spiritu', 'spjanuari',
'spk', 'spl', 'splash', 'splashmobil', 'splat', 'splendid', 'split', 'splle', 'splwat', 'spoil', '
spoilt', 'spoke', 'spoken', 'sponsor', 'spontan', 'spook', 'spoon', 'sporad', 'sport', 'sportsx',
'spose', 'spot', 'spotti', 'spous', 'sppok', 'spreadsheet', 'spree', 'spring', 'sprint', 'sprwm',
'sptv', 'sptyron', 'spunout', 'sq825', 'squat', 'squeeeeez', 'squeez', 'squid', 'squishi', 'sr', '
sri', 'srsli', 'srt', 'ssi', 'ssindia', 'ssnervou', 'st', 'stabil', 'stabl', 'stadium', 'staff', '
staffsciencenusedusgphyhcmkteachingpc1323', 'stage', 'stagwood', 'stair', 'stalk', 'stamp', 'stand
', 'standard', 'stapati', 'star', 'stare', 'starer', 'starshin', 'start', 'startedindia',
'starti', 'starv', 'starwars3', 'stash', 'state', 'statement', 'station', 'statu', 'stay', 'stayin
', 'std', 'stdtxtrate', 'steak', 'steal', 'steam', 'steamboat', 'steed', 'steer', 'step', 'stereo'
, 'stereophon', 'sterl', 'sterm', 'steve', 'stevelik', 'stewarts', 'steyn', 'sth', 'sthi', 'stick'
, 'sticki', 'stifl', 'stil', 'still', 'stillmayb', 'stink', 'stitch', 'stock', 'stockport', 'stole
n', 'stomach', 'stomp', 'stone', 'stoner', 'stool', 'stop', 'stop2', 'stop2stop', 'stopbcm', 'stop
c', 'stopcost', 'stoptxt', 'stoptxtstop', 'store', 'storelik', 'stori', 'storm', 'str', 'str8', 's
traight', 'strain', 'strang', 'stranger', 'strangersaw', 'stream', 'street', 'streetshal',
'stress', 'stressful', 'stretch', 'strewn', 'strict', 'strike', 'string', 'strip', 'stripe', 'stro
ke', 'strong', 'strongbuy', 'strongli', 'strt', 'strtd', 'struggl', 'stu', 'stubborn', 'stuck', 's
tuddi', 'student', 'studentfinanci', 'studentsthi', 'studi', 'studio', 'studyn', 'stuf', 'stuff',
'stuff42moro', 'stuffleav', 'stuffwhi', 'stun', 'stupid', 'stupidit', 'style', 'stylish',
'stylist', 'sub', 'subject', 'sublet', 'submit', 'subpoli', 'subscrib', 'subscribe6gbpmnth',
'subscript', 'subscriptn3gbpwk', 'subscrit', 'subsequ', 'subtoitl', 'success', 'such', 'suck', 'su
cker', 'sudden', 'suddenli', 'sudn', 'sue', 'suffer', 'suffici', 'sugabab', 'suganya', 'sugar', 's
ugardad', 'suggest', 'suit', 'suitem', 'sullivan', 'sum', 'sum1', 'sumf', 'summer', 'summon', 'sum
thin', 'sumthinxx', 'sun', 'sun0819', 'sunday', 'sundayish', 'sunlight', 'sunni', 'sunoco',
'sunroof', 'sunscreen', 'sunshin', 'suntec', 'sup', 'super', 'superb', 'superior', 'supervisor', '
supli', 'supos', 'suppli', 'supplier', 'support', 'supportprovid', 'supportveri', 'suppos', 'supre
m', 'suprman', 'sura', 'sure', 'surf', 'surgic', 'surli', 'surnam', 'surpris', 'surrend',
'surround', 'survey', 'surya', 'sutra', 'sux', 'suzi', 'svc', 'sw7', 'sw73ss', 'swalpa', 'swan', '
swann', 'swap', 'swashbuckl', 'swat', 'swatch', 'sway', 'swayz', 'swear', 'sweater', 'sweatter', '
sweet', 'sweetest', 'sweetheart', 'sweeti', 'swell', 'swhrt', 'swim', 'swimsuit', 'swing', 'swiss'
, 'switch', 'swollen', 'swoop', 'swt', 'swtheart', 'syd', 'syllabu', 'symbol', 'sympathet',
'symptom', 'sync', 'syria', 'syrup', 'system', 't91', 'ta', 'tabl', 'tablet', 'tackl', 'taco', 'ta
ct', 'tactless', 'tadaaaaa', 'tag', 'tahan', 'tai', 'tait', 'taj', 'taka', 'take', 'takecar', 'tak
en', 'takenonli', 'takin', 'talent', 'talk', 'talkbut', 'talkin', 'tall', 'tallahasse', 'tallent',
'tamilnaduthen', 'tampa', 'tank', 'tantrum', 'tap', 'tape', 'tariff', 'tarot', 'tarpon', 'tast', '
tat', 'tata', 'tattoo', 'tau', 'taught', 'taunton', 'tax', 'taxi', 'taxless', 'taxt', 'taylor', 't
ayseertissco', 'tb', 'tbspersolvo', 'tc', 'tcllc', 'tcrw1', 'tcsbcm4235wc1n3xx', 'tcsc',
'tcsstop', 'tddnewsletteremc1couk', 'tea', 'teach', 'teacher', 'teacoffe', 'team', 'tear', 'teas',
'tech', 'technic', 'technolog', 'tee', 'teenag', 'teeth', 'teethi', 'teethif', 'teju', 'tel', 'tel
ephon', 'teletext', 'tell', 'telli', 'tellmiss', 'telphon', 'telugu', 'telugutht', 'temal', 'temp'
, 'temper', 'templ', 'ten', 'tenant', 'tendenc', 'tenerif', 'tens', 'tension', 'teresa', 'term', '
terminatedw', 'termsappli', 'terri', 'terribl', 'terrif', 'terrorist', 'tesco', 'tessypl', 'test',
'tex', 'texa', 'texd', 'text', 'text82228', 'textand', 'textbook', 'textbuddi', 'textcomp',
'textin', 'textoper', 'textpod', 'textsweekend', 'tgxxrz', 'th', 'than', 'thandiyachu', 'thangam',
'thangamit', 'thank', 'thanks2', 'thanksgiv', 'thanku', 'thankyou', 'thanx', 'thanx4', 'thanxxx',
'thasa', 'that', 'that2worzel', 'thatd', 'thatdont', 'thati', 'thatll', 'thatmum', 'thatnow', 'the
', 'the4th', 'theacus', 'theater', 'theatr', 'thecd', 'thedailydraw', 'thekingshead', 'them',
'theme', 'themob', 'themobhit', 'themobyo', 'themp', 'then', 'thenwil', 'theoret', 'theori',
'theplac', 'thepub', 'there', 'theredo', 'theregoodnight', 'therel', 'therer', 'therexx', 'these',
'theseday', 'theseyour', 'thesi', 'thesmszonecom', 'thewend', 'they', 'theyll', 'theyr', 'thgt', '
thi', 'thia', 'thin', 'thing', 'thinghow', 'think', 'thinkin', 'thinkthi', 'thinl',
'thirunelvali', 'thisdon', 'thk', 'thkin', 'thm', 'thnk', 'thnq', 'thnx', 'tho', 'those', 'thoso',
'thot', 'thou', 'though', 'thought', 'thoughtsi', 'thousand', 'thout', 'thread', 'threat',
'three', 'threw', 'thriller', 'throat', 'throw', 'throwin', 'thrown', 'thru', 'thrurespect', 'tht'
, 'thu', 'thuglyf', 'thur', 'thursday', 'thx', 'ti', 'tick', 'ticket', 'tiempo', 'tiger', 'tight',
'tightli', 'tigress', 'tih', 'tiim', 'til', 'till', 'tim', 'time', 'timedhoni', 'timegud',
'timehop', 'timeslil', 'timey', 'timeyour', 'timi', 'timin', 'tini', 'tip', 'tire', 'tirunelvai',
'tirunelvali', 'tirupur', 'tisscotays', 'titl', 'titleso', 'tiwari', 'tix', 'tiz', 'tke', 'tkt', '
tlk', 'tm', 'tming', 'tmobil', 'tmorrowpl', 'tmr', 'tmrw', 'tmw', 'tnc', 'toa', 'toaday',
'tobacco', 'tobe', 'tocallshal', 'toclaim', 'today', 'todaybut', 'todaydo', 'todayfrom',
'todaygood', 'todayh', 'todaysundaysunday', 'todo', 'tog', 'togeth', 'tohar', 'toilet', 'tok', 'to
ke', 'token', 'tol', 'told', 'toldsh', 'toledo', 'toler', 'toleratbc', 'toll', 'tom', 'tomarrow',
'tome', 'tomeandsaidthi', 'tomo', 'tomoc', 'tomorro', 'tomorrow', 'tomorrowcal', 'tomorrowtoday',
'tomorw', 'ton', 'tone', 'tones2u', 'tones2youcouk', 'tonesrepli', 'tonex', 'tonght', 'tongu', 'to
night', 'tonit', 'tonitebusi', 'toniteth', 'tonsolitusaswel', 'too', 'took', 'tookplac', 'tool', '
toolet', 'tooo', 'toopray', 'toot', 'toothpast', 'tootsi', 'top', 'topic', 'topicsorri', 'toplay',
'toppoli', 'tor', 'torch', 'torrent', 'tortilla', 'tortur', 'tosend', 'toshiba', 'toss', 'tot', 't
otal', 'tote', 'touch', 'tough', 'toughest', 'tour', 'toward', 'town', 'towncud', 'towndontmatt',
'toxic', 'toyota', 'tp', 'track', 'trackmarqu', 'trade', 'tradit', 'traffic', 'train', 'trainner',
'tram', 'tranquil', 'transact', 'transcrib', 'transfer', 'transferacc', 'transfr', 'transport', 't
rash', 'trauma', 'trav', 'travel', 'treacl', 'treadmil', 'treasur', 'treat', 'treatin', 'trebl', '
tree', 'trek', 'trend', 'tri', 'trial', 'trip', 'tripl', 'trishul', 'triumph', 'tron', 'troubl', '
troubleshoot', 'trouser', 'trubl', 'truck', 'true', 'truekdo', 'truffl', 'truli', 'truro',
'trust', 'truth', 'tryin', 'trywal', 'ts', 'tsandc', 'tsc', 'tscs08714740323',
'tscs087147403231winawkage16', 'tshirt', 'tsunami', 'tt', 'ttyl', 'tue', 'tuesday', 'tui',
'tuition', 'tul', 'tulip', 'tund', 'tune', 'tunji', 'turkey', 'turn', 'tuth', 'tv', 'tvhe',
'tvlol', 'twat', 'twelv', 'twenti', 'twice', 'twigg', 'twilight', 'twin', 'twink', 'twitter', 'two
', 'txt', 'txt250com', 'txtauction', 'txtauctiontxt', 'txtin', 'txting', 'txtjourney', 'txtno', 't
xtx', 'tyler', 'type', 'typelyk', 'typic', 'u', 'u2moro', 'uawakefeellikw', 'ubandu', 'ubi', 'ucal
', 'ufind', 'ugadi', 'ugh', 'ugo', 'uh', 'uhhhhrmm', 'uif', 'uin', 'ujhhhhhhh', 'uk', 'ukmobiled',
'ukp2000', 'ull', 'ultim', 'ultimatum', 'um', 'umma', 'ummmawil', 'ummmmmaah', 'un', 'unabl', 'unb
eliev', 'uncl', 'unclaim', 'uncomfort', 'uncondit', 'unconsci', 'unconvinc', 'uncount', 'uncut', '
under', 'underdtand', 'understand', 'understood', 'underwear', 'undrstnd', 'undrstndng',
'unemploy', 'unev', 'unfold', 'unfortun', 'unfortuntli', 'unhappi', 'uni', 'unicef', 'uniform', 'u
nintent', 'uniqu', 'uniquei', 'unit', 'univ', 'univers', 'unknown', 'unless', 'unlik', 'unlimit',
'unmit', 'unnecessarili', 'unni', 'unrecogn', 'unredeem', 'unsecur', 'unsold', 'unsoldmik',
'unsoldnow', 'unspoken', 'unsub', 'unsubscrib', 'until', 'unusu', 'uothrwis', 'up', 'up4', 'upchar
g', 'upd8', 'updat', 'updatenow', 'upgrad', 'upgrdcentr', 'uphad', 'upload', 'upnot', 'upon', 'ups
et', 'upseti', 'upsetit', 'upstair', 'upto', 'uptown', 'upyeh', 'ur', 'ure', 'urfeel', 'urgent', '
urgentbut', 'urgentlyit', 'urgh', 'urgnt', 'urgoin', 'urgran', 'urin', 'url', 'urmomi', 'urn', 'ur
self', 'us', 'usb', 'usc', 'uscedu', 'use', 'useless', 'user', 'usf', 'usget', 'usher', 'uslet', '
usml', 'usno', 'uso', 'usp', 'usual', 'usualiam', 'uteru', 'utter', 'uup', 'uv', 'uve', 'uwana', '
uwant', 'uworld', 'uxxxx', 'v', 'vaazhthukk', 'vagu', 'vai', 'vale', 'valentin', 'valid',
'valid12hr', 'valu', 'valuabl', 'valuemorn', 'varaya', 'vargu', 'vari', 'variou', 'varma',
'vasai', 'vat', 'vatian', 'vava', 'vco', 'vday', 'vega', 'veget', 'veggi', 'vehicl', 'velacheri',
'velli', 'velusami', 'venaam', 'venugop', 'veri', 'verifi', 'version', 'versu', 'vettam', 'vewi',
'via', 'vibrant', 'vibrat', 'vic', 'victor', 'victoria', 'vid', 'video', 'videochat', 'videop', 'v
ideophon', 'videosound', 'videosounds2', 'vidnot', 'view', 'vijay', 'vijaykanth', 'vikki',
'vikkyim', 'vilikkamt', 'vill', 'villa', 'villag', 'vinobanagar', 'violat', 'violenc', 'violet', '
vip', 'virgil', 'virgin', 'virtual', 'visa', 'visionsmscom', 'visit', 'visitne', 'visitor',
'vital', 'vitamin', 'viva', 'vivek', 'vivekanand', 'viveki', 'vl', 'vldo', 'voda', 'vodafon',
'vodka', 'voic', 'voicemail', 'voila', 'volcano', 'vomit', 'vomitin', 'vote', 'voucher',
'voucherstext', 'vpist', 'vpod', 'vri', 'vs', 'vth', 'vtire', 'w', 'w111wx', 'w14rg', 'w1a', 'w1j'
, 'w1t1ji', 'w45wq', 'w8in', 'wa', 'wa14', 'waaaat', 'wad', 'wadebridgei', 'wah', 'wahala',
'wahay', 'wahe', 'waheeda', 'wahleykkumshar', 'waht', 'wait', 'waiti', 'waitin', 'waitshould',
'waitu', 'wake', 'wale', 'walik', 'walk', 'walkabout', 'walkin', 'wall', 'wallet', 'wallpap',
'wallpaperal', 'walmart', 'walsal', 'wamma', 'wan', 'wan2', 'wana', 'wanna', 'wannatel', 'want', '
want2com', 'wap', 'waqt', 'warm', 'warn', 'warner', 'warranti', 'warwick', 'washob', 'wasnt', 'was
t', 'wat', 'watch', 'watchin', 'watchng', 'wate', 'water', 'watev', 'watevr', 'watll',
'watrdayno', 'watt', 'wave', 'way', 'way2smscom', 'waythi', 'wc', 'wc1n', 'wc1n3xx', 'weak',
'weapon', 'wear', 'weasel', 'weather', 'web', 'web2mobil', 'webadr', 'webeburnin', 'webpag',
'websit', 'websitenow', 'wed', 'weddin', 'weddingfriend', 'wedlunch', 'wednesday', 'wee', 'weed',
'weeddefici', 'week', 'weekday', 'weekend', 'weekli', 'weekstop', 'weigh', 'weight', 'weighthaha',
'weightloss', 'weird', 'weirdest', 'weirdi', 'weirdo', 'weiyi', 'welcom', 'well', 'wellda', 'welli
', 'welltak', 'wellyou', 'welp', 'wen', 'wendi', 'wenev', 'went', 'wenwecan', 'wer', 'were', 'were
ar', 'werebor', 'werent', 'wereth', 'wesley', 'west', 'western', 'westlif', 'westonzoyland',
'westshor', 'wet', 'wetherspoon', 'weve', 'wewa', 'whassup', 'what', 'whatev', 'whatsup', 'wheat',
'wheel', 'wheellock', 'when', 'whenev', 'whenevr', 'whenr', 'whenwher', 'where', 'wherear',
'wherebtw', 'wherev', 'wherevr', 'wherr', 'whether', 'whi', 'which', 'while', 'whileamp',
'whilltak', 'whisper', 'white', 'whn', 'who', 'whole', 'whom', 'whore', 'whose', 'whr', 'wi', 'wic
k', 'wicket', 'wicklow', 'wid', 'widelivecomindex', 'wif', 'wife', 'wifedont', 'wifehow', 'wifi',
'wihtuot', 'wikipediacom', 'wil', 'wild', 'wildest', 'wildlif', 'will', 'willpow', 'win',
'win150ppmx3age16', 'wind', 'windi', 'window', 'wine', 'wing', 'winner', 'winnersclub',
'winterston', 'wipe', 'wipro', 'wiproy', 'wire3net', 'wisdom', 'wise', 'wish', 'wishin',
'wishlist', 'wiskey', 'wit', 'with', 'withdraw', 'wither', 'within', 'without', 'witin', 'witot',
'witout', 'wiv', 'wizzl', 'wk', 'wkend', 'wkent150p16', 'wkg', 'wkli', 'wknd', 'wktxt', 'wlcome',
'wld', 'wmlid1b6a5ecef91ff937819firsttrue180430jul05', 'wmlid820554ad0a1705572711firsttru',
'wnevr', 'wnt', 'wo', 'woah', 'wocay', 'woke', 'woken', 'woman', 'womdarful', 'women', 'won', 'won
dar', 'wondarful', 'wonder', 'wont', 'woo', 'wood', 'woodland', 'woohoo', 'woot', 'woould',
'woozl', 'worc', 'word', 'wordcollect', 'wordnot', 'wordsevri', 'wordstart', 'work', 'workag',
'workand', 'workin', 'worklov', 'workout', 'world', 'worldgnun', 'worldmay', 'worldveri', 'worm',
'worri', 'worriedx', 'worryc', 'worryus', 'wors', 'worst', 'worth', 'worthless', 'wot', 'wotu', 'w
otz', 'woul', 'would', 'woulda', 'wouldnt', 'wound', 'wow', 'wquestion', 'wrc', 'wreck', 'wrench',
'wright', 'write', 'writh', 'wrk', 'wrki', 'wrkin', 'wrking', 'wrld', 'wrnog', 'wrong', 'wrongli',
'wrongtak', 'wrote', 'ws', 'wt', 'wtc', 'wtf', 'wth', 'wthout', 'wud', 'wudnt', 'wuld', 'wuldnt',
'wun', 'www07781482378com', 'www4tcbiz', 'www80488biz', 'wwwapplausestorecom',
'wwwareyouuniquecouk', 'wwwasjesuscom', 'wwwb4utelecom', 'wwwbridalpetticoatdreamscouk',
'wwwcashbincouk', 'wwwclubmobycom', 'wwwclubzedcouk', 'wwwcnupdatescomnewslett', 'wwwcomuknet', 'w
wwdbuknet', 'wwwflirtpartyu', 'wwwfullonsmscom', 'wwwgambtv', 'wwwgetzedcouk', 'wwwidewcom',
'wwwldewcom', 'wwwldewcom1win150ppmx3age16', 'wwwldewcom1win150ppmx3age16subscript',
'wwwldewcomsubs161win150ppmx3', 'wwwmovietriviatv', 'wwwmusictrivianet', 'wwworangecoukow',
'wwwphb1com', 'wwwregalportfoliocouk', 'wwwringtonekingcouk', 'wwwringtonescouk',
'wwwrtfsphostingcom', 'wwwsantacallingcom', 'wwwshortbreaksorguk', 'wwwsmsacubootydeli',
'wwwsmsacugoldvik', 'wwwsmsacuhmmross', 'wwwsmsacunat27081980', 'wwwsmsacunatalie2k9',
'wwwsmsconet', 'wwwtcbiz', 'wwwtelediscountcouk', 'wwwtextcompcom', 'wwwtextpodnet', 'wwwtklscom',
'wwwtxt2shopcom', 'wwwtxt43com', 'wwwtxt82228com', 'wwwtxttowincouk', 'wwwwin82050couk', 'wyli', '
x', 'x29', 'x49', 'x49your', 'xafter', 'xam', 'xavier', 'xchat', 'xclusiveclubsaisai', 'xin', 'xma
', 'xnet', 'xoxo', 'xt', 'xuhui', 'xx', 'xxsp', 'xxuk', 'xxx', 'xxxmobilemovieclub',
'xxxmobilemovieclubcomnqjkgighjjgcbl', 'xxxx', 'xxxxx', 'xxxxxx', 'xxxxxxx', 'xxxxxxxx',
'xxxxxxxxxxxxxx', 'xy', 'y87', 'ya', 'yago', 'yah', 'yahoo', 'yalrigu', 'yalru', 'yam', 'yan', 'ya
r', 'yard', 'yavnt', 'yaxx', 'yaxxx', 'yay', 'yck', 'yday', 'ye', 'yeah', 'yeahand', 'year',
'yeesh', 'yeh', 'yell', 'yellow', 'yelowi', 'yen', 'yeovil', 'yep', 'yer', 'yes165', 'yes434', 'ye
s440', 'yes762', 'yes910', 'yesbut', 'yesfrom', 'yesgauti', 'yesh', 'yesher', 'yesim', 'yesmum', '
yessura', 'yest', 'yesterday', 'yet', 'yetti', 'yetund', 'yi', 'yifeng', 'yiju',
'yijuehotmailcom', 'ym', 'ymca', 'yo', 'yoga', 'yogasana', 'yoher', 'yor', 'yorg', 'you',
'youani', 'youcarlo', 'youclean', 'youd', 'youdearwith', 'youdo', 'youhow', 'youi', 'youkwher', 'y
ould', 'youll', 'youmi', 'youmoney', 'young', 'younger', 'youphon', 'your', 'yourinclus',
'yourjob', 'yourself', 'youso', 'youthat', 'youto', 'youuuuu', 'youv', 'youwanna', 'youwhen', 'yov
il', 'yowif', 'yoyyooo', 'yr', 'ystrdayic', 'yummi', 'yummmm', 'yun', 'yunni', 'yuo', 'yuou', 'yup
', 'yupz', 'ywhere', 'zac', 'zaher', 'zealand', 'zebra', 'zed', 'zero', 'zhong', 'zindgi', 'zoe',
'zogtoriu', 'zoom', 'zouk', 'zyada', 'Ü', 'é', 'ü', 'üll', '亖ud']
```

In [125]:

```
data_sample=data[0:10]

cv2=CountVectorizer(analyzer=clean_text)

X=cv2.fit_transform(data_sample['msg'])
X.shape
```

Out[125]:

```
(10, 131)
```

In [127]:

```
df=pd.DataFrame(X.toarray(), columns=cv2.get_feature_names())
df
```

```
df
```

Out[127]:

	08002986030	08452810075over18	09061701461	11	12	150	2	2005	21st	3	...	vettam	wat	week	wif	win	winner	wkli	worc
0	0	0	0	0	0	0	0	0	0	0	...	0	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	1	0	0	0	0
2	0	1	0	0	0	0	1	1	1	0	...	0	0	0	0	1	0	1	0
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0	1	...	0	0	1	0	0	0	0	1
6	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	...	1	0	0	0	0	0	0	0
8	0	0	1	0	1	0	0	0	0	0	...	0	0	0	0	0	1	0	0
9	1	0	0	1	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0

10 rows × 131 columns

N-Grams

N-Grams creates a document-term matrix,

- columns represent all columns of adjacent words of length n
- cells represent count

*--> Eg: "I Am studying NLP"

```
1. bigram : 'I am', 'am studying', 'studying NLP'===> group of 2 words
2. trigram:'I am studying', 'am studying NLP' ===> group of 3 words
3. 4-gram : 'I am studying NLP' ===> group of 4 words
```

Read Raw Text

In [130]:

```python
import nltk
import string
import re
import pandas as pd

stopwords=nltk.corpus.stopwords.words('english')
ps=PorterStemmer()

data=pd.read_csv('SMSSpamCollection', sep='\t', header=None, names=['label', 'msg'])
data.head()
```

Out[130]:

	label	msg
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives around here though

Text Cleaning

In [133]:

```
def clean_text(txt):
    text=''.join([c for c in txt if c not in string.punctuation])
    tokens=re.split('\W+', text)
    text_nostop=' '.join([ps.stem(word) for word in tokens if word not in stopwords])
    return text_nostop

data['msg_clean']=data['msg'].apply(lambda x: clean_text(x))

data.head()
```

Out[133]:

	label	msg	msg_clean
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	Go jurong point crazi avail bugi n great world la e buffet cine got amor wat
1	ham	Ok lar... Joking wif u oni...	Ok lar joke wif u oni
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	free entri 2 wkli comp win FA cup final tkt 21st may 2005 text FA 87121 receiv entri questionstd...
3	ham	U dun say so early hor... U c already then say...	U dun say earli hor U c alreadi say
4	ham	Nah I don't think he goes to usf, he lives around here though	nah I dont think goe usf live around though

In [140]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer(ngram_range=(2,2))

corpus=['This is a sentence is',
       'this is another sentence',
       'third document is here']

X=cv.fit_transform(corpus)
print(X.shape)
print('='*50)
print(X)
print('='*50)
print(X.toarray())
print('='*50)
print(cv.vocabulary_)
print('='*50)
print(cv.get_feature_names())
```

```
(3, 8)
==================================================
  (0, 5)	1
  (0, 4)	1
  (0, 7)	1
  (1, 0)	1
  (1, 2)	1
  (1, 7)	1
  (2, 3)	1
  (2, 1)	1
  (2, 6)	1
==================================================
[[0 0 0 0 1 1 0 1]
 [1 0 1 0 0 0 0 1]
 [0 1 0 1 0 0 1 0]]
==================================================
{'this is': 7, 'is sentence': 4, 'sentence is': 5, 'is another': 2, 'another sentence': 0, 'third
document': 6, 'document is': 1, 'is here': 3}
==================================================
['another sentence', 'document is', 'is another', 'is here', 'is sentence', 'sentence is', 'third
document', 'this is']
```

In [142]:

```
df=pd.DataFrame(X.toarray(), columns=cv.get_feature_names())
df
```

Out[142]:

	another sentence	document is	is another	is here	is sentence	sentence is	third document	this is
0	0	0	0	0	1	1	0	1
1	1	0	1	0	0	0	0	1
2	0	1	0	1	0	0	1	0

Bi and Trigram

ngram_range=(2,3)

In [143]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer(ngram_range=(2,3))

corpus=['This is a sentence is',
       'this is another sentence',
       'third document is here']

X=cv.fit_transform(corpus)
print(X.shape)
print('='*50)
print(X)
print('='*50)
print(X.toarray())
print('='*50)
print(cv.vocabulary_)
print('='*50)
print(cv.get_feature_names())
```

```
(3, 14)
==================================================
  (0, 7)	1
  (0, 13)	1
  (0, 8)	1
  (0, 6)	1
  (0, 11)	1
  (1, 4)	1
  (1, 12)	1
  (1, 0)	1
  (1, 3)	1
  (1, 11)	1
  (2, 2)	1
  (2, 10)	1
  (2, 5)	1
  (2, 1)	1
  (2, 9)	1
==================================================
[[0 0 0 0 0 0 1 1 1 0 0 1 0 1]
 [1 0 0 1 1 0 0 0 0 0 0 1 1 0]
 [0 1 1 0 0 1 0 0 0 1 1 0 0 0]]
==================================================
{'this is': 11, 'is sentence': 6, 'sentence is': 8, 'this is sentence': 13, 'is sentence is': 7, '
is another': 3, 'another sentence': 0, 'this is another': 12, 'is another sentence': 4, 'third doc
ument': 9, 'document is': 1, 'is here': 5, 'third document is': 10, 'document is here': 2}
==================================================
['another sentence', 'document is', 'document is here', 'is another', 'is another sentence', 'is h
ere', 'is sentence', 'is sentence is', 'sentence is', 'third document', 'third document is', 'this
is', 'this is another', 'this is sentence']
```

In [144]:

```
df=pd.DataFrame(X.toarray(), columns=cv.get_feature_names())
df
```

Out[144]:

	another sentence	document is	document is here	is another	is another sentence	is here	is sentence	is sentence is	sentence is	third document	third document is	this is	this is another	thi sente
0	0	0	0	0	0	0	1	1	1	0	0	1	0	
1	1	0	0	1	1	0	0	0	0	0	0	1	1	
2	0	1	1	0	0	1	0	0	0	1	1	0	0	

In [146]:

```
### CountVectorization on SMSSpamCollection
from sklearn.feature_extraction.text import CountVectorizer
cv1=CountVectorizer(ngram_range=(2,3))

X=cv1.fit_transform(data['msg_clean'])
print(X.shape)
```

```
(5572, 69437)
```

In [147]:

```
data_sample=data[0:10]
cv2=CountVectorizer(ngram_range=(2,3))

X=cv2.fit_transform(data_sample['msg_clean'])
X.shape
```

Out[147]:

```
(10, 242)
```

In [149]:

```
df=pd.DataFrame(X.toarray(), columns=cv2.get_feature_names())
df
```

Out[149]:

	09061701461 claim	09061701461 claim code	11 month	11 month entitl	12 hour	150 rcv	2005 text	2005 text fa	21st may	21st may 2005	...	winner as	winner as valu	wkli comp	wkli comp win	word back	word back id	wor
0	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
2	0	0	0	0	0	0	1	1	1	1	...	0	0	1	1	0	0	
3	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
5	0	0	0	0	0	1	0	0	0	0	...	0	0	0	0	1	1	
6	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
7	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	
8	1	1	0	0	1	0	0	0	0	0	...	1	1	0	0	0	0	
9	0	0	1	1	0	0	0	0	0	0	...	0	0	0	0	0	0	

10 rows × 242 columns

TF-IDF

- Term Frequency- Inverse Document Frequency

Creates document term matrix

- columns are individual unique words

- cells contain a weight which signifies how important a word is for an individual text message

Wi,j=TFi,j * log(N/dfi)

- **Term Frequency(TF)** : TFi,j==> number of times term i occurs in jth document divided by number of terms in j
- **Inverse Document Frequency(TF)** : log(N/dfi)= total number of documents/no. of documents containing i

=> Example:

j : " I am studying NLP"

tf(am,j)=1/4=0.25

N=200

df(am)=2

w(am,j)=TFi,j *log(N/dfi) = 0.25* (200/2) =0.25*2 =0.5

Raw Text

In [153]:

```
import nltk
import re
import string
import pandas as pd

stopwords=nltk.corpus.stopwords.words('english')
ps=PorterStemmer()

data=pd.read_csv('SMSSpamCollection', sep='\t', header=None, names=['label', 'msg'])
data.head()
```

Out[153]:

	label	msg
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives around here though

Clean Text

In [154]:

```
def clean_text(txt):
    text=''.join([c for c in txt if c not in string.punctuation])
    tokens=re.split('\W+', text)
    text_nostop=[ps.stem(word) for word in tokens if word not in stopwords]
    return text_nostop
```

In [ ]:

```
#data['msg_clean']=data['msg'].apply(lambda x: clean_text())
```

TFIDF Vectorizer

In [156]:

```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf_vec=TfidfVectorizer()

corpus=['This is a sentence is',
       'this is another sentence',
       'third document is here']

X=tfidf_vec.fit_transform(corpus)

print(X.shape)
print(tfidf_vec.vocabulary_)
print('='*50)
print(tfidf_vec.get_feature_names())
print(X)
print('='*50)
print(X.toarray())
```

```
(3, 7)
{'this': 6, 'is': 3, 'sentence': 4, 'another': 0, 'third': 5, 'document': 1, 'here': 2}
==================================================
['another', 'document', 'here', 'is', 'sentence', 'third', 'this']
  (0, 6)	0.47606293927679294
  (0, 3)	0.7394106813498715
  (0, 4)	0.47606293927679294
  (1, 6)	0.4804583972923858
  (1, 3)	0.3731188059313277
  (1, 4)	0.4804583972923858
  (1, 0)	0.6317450542765208
  (2, 3)	0.3227445421804912
  (2, 5)	0.546454011634009
  (2, 1)	0.546454011634009
  (2, 2)	0.546454011634009
==================================================
[[0.         0.         0.         0.73941068 0.47606294 0.
  0.47606294]
 [0.63174505 0.         0.         0.37311881 0.4804584  0.
  0.4804584 ]
 [0.         0.54645401 0.54645401 0.32274454 0.         0.54645401
  0.        ]]
```

In [158]:

```
df=pd.DataFrame(X.toarray(), columns=tfidf_vec.get_feature_names())
df
```

Out[158]:

	another	document	here	is	sentence	third	this
0	0.000000	0.000000	0.000000	0.739411	0.476063	0.000000	0.476063
1	0.631745	0.000000	0.000000	0.373119	0.480458	0.000000	0.480458
2	0.000000	0.546454	0.546454	0.322745	0.000000	0.546454	0.000000

TFIDF Vectorization on SMSspamcollection

In [159]:

```
tfidf1=TfidfVectorizer(analyzer=clean_text)

X=tfidf1.fit_transform(data['msg'])
X.shape
```

Out[159]:

```
(5572, 8340)
```

In [161]:

```
data_sample=data[0:10]
```

```
tfidf3=TfidfVectorizer(analyzer=clean_text)

X=tfidf3.fit_transform(data_sample['msg'])

print(X.shape)
print('='*50)
print(tfidf3.vocabulary_)
print('='*50)
print(tfidf3.get_feature_names())
print('='*50)
print(X)
print('='*50)
print(X.toarray())
```

```
(10, 131)
==================================================
{'Go': 16, 'jurong': 70, 'point': 91, 'crazi': 46, 'avail': 29, 'bugi': 33, 'n': 82, 'great': 64,
'world': 129, 'la': 72, 'e': 52, 'buffet': 32, 'cine': 40, 'got': 63, 'amor': 26, 'wat': 122, 'Ok'
: 19, 'lar': 73, 'joke': 69, 'wif': 124, 'u': 116, 'oni': 87, 'free': 58, 'entri': 55, '2': 6, 'wk
li': 127, 'comp': 44, 'win': 125, 'FA': 15, 'cup': 47, 'final': 57, 'tkt': 113, '21st': 8, 'may':
77, '2005': 7, 'text': 108, '87121': 10, 'receiv': 97, 'questionstd': 94, 'txt': 115, 'ratetc': 95
, 'appli': 27, '08452810075over18': 1, 'U': 23, 'dun': 51, 'say': 101, 'earli': 53, 'hor': 67, 'c'
: 34, 'alreadi': 25, 'nah': 83, 'I': 17, 'dont': 50, 'think': 111, 'goe': 62, 'usf': 118, 'live':
76, 'around': 28, 'though': 112, 'freemsg': 59, 'hey': 66, 'darl': 49, '3': 9, 'week': 123, 'word'
: 128, 'back': 30, 'Id': 18, 'like': 75, 'fun': 61, 'still': 107, 'Tb': 21, 'ok': 86, 'xxx': 130,
'std': 106, 'chg': 39, 'send': 103, '150': 5, 'rcv': 96, 'even': 56, 'brother': 31, 'speak': 105,
'they': 110, 'treat': 114, 'aid': 24, 'patent': 89, 'As': 13, 'per': 90, 'request': 99, 'mell': 78
, 'oru': 88, 'minnaminungint': 79, 'nurungu': 85, 'vettam': 121, 'set': 104, 'callertun': 37, 'cal
ler': 36, 'press': 92, '9': 11, 'copi': 45, 'friend': 60, 'winner': 126, 'valu': 120, 'network': 8
4, 'custom': 48, 'select': 102, 'receivea': 98, '900': 12, 'prize': 93, 'reward': 100, 'To': 22, '
claim': 41, 'call': 35, '09061701461': 2, 'code': 42, 'kl341': 71, 'valid': 119, '12': 4, 'hour':
68, 'had': 65, 'mobil': 80, '11': 3, 'month': 81, 'R': 20, 'entitl': 54, 'updat': 117, 'latest': 7
4, 'colour': 43, 'camera': 38, 'the': 109, 'Co': 14, '08002986030': 0}
==================================================
['08002986030', '08452810075over18', '09061701461', '11', '12', '150', '2', '2005', '21st', '3', '
87121', '9', '900', 'As', 'Co', 'FA', 'Go', 'I', 'Id', 'Ok', 'R', 'Tb', 'To', 'U', 'aid',
'alreadi', 'amor', 'appli', 'around', 'avail', 'back', 'brother', 'buffet', 'bugi', 'c', 'call', '
caller', 'callertun', 'camera', 'chg', 'cine', 'claim', 'code', 'colour', 'comp', 'copi', 'crazi',
'cup', 'custom', 'darl', 'dont', 'dun', 'e', 'earli', 'entitl', 'entri', 'even', 'final', 'free',
'freemsg', 'friend', 'fun', 'goe', 'got', 'great', 'had', 'hey', 'hor', 'hour', 'joke', 'jurong',
'kl341', 'la', 'lar', 'latest', 'like', 'live', 'may', 'mell', 'minnaminungint', 'mobil', 'month',
'n', 'nah', 'network', 'nurungu', 'ok', 'oni', 'oru', 'patent', 'per', 'point', 'press', 'prize',
'questionstd', 'ratetc', 'rcv', 'receiv', 'receivea', 'request', 'reward', 'say', 'select',
'send', 'set', 'speak', 'std', 'still', 'text', 'the', 'they', 'think', 'though', 'tkt', 'treat',
'txt', 'u', 'updat', 'usf', 'valid', 'valu', 'vettam', 'wat', 'week', 'wif', 'win', 'winner', 'wkl
i', 'word', 'world', 'xxx']
==================================================
  (0, 16)	0.25000000000000006
  (0, 70)	0.25000000000000006
  (0, 91)	0.25000000000000006
  (0, 46)	0.25000000000000006
  (0, 29)	0.25000000000000006
  (0, 33)	0.25000000000000006
  (0, 82)	0.25000000000000006
  (0, 64)	0.25000000000000006
  (0, 129)	0.25000000000000006
  (0, 72)	0.25000000000000006
  (0, 52)	0.25000000000000006
  (0, 32)	0.25000000000000006
  (0, 40)	0.25000000000000006
  (0, 63)	0.25000000000000006
  (0, 26)	0.25000000000000006
  (0, 122)	0.25000000000000006
  (1, 19)	0.408248290463863
  (1, 73)	0.408248290463863
  (1, 69)	0.408248290463863
  (1, 124)	0.408248290463863
  (1, 116)	0.408248290463863
  (1, 87)	0.408248290463863
  (2, 58)	0.1644469402151285
  (2, 55)	0.38689239288182964
  (2, 6)	0.19344619644091482
  :	:
  (8, 22)	0.21594036407575087
  (8, 41)	0.43188072815150175
  (8, 35) 0.18356903777141104
  (8, 2) 0.21594036407575087
  (8, 42) 0.21594036407575087
  (8, 71) 0.21594036407575087
  (8, 119) 0.21594036407575087
  (8, 4) 0.21594036407575087
  (8, 68) 0.21594036407575087
  (9, 58) 0.31939406929606784
  (9, 23) 0.15969703464803392
  (9, 35) 0.15969703464803392
  (9, 65) 0.1878586728043793
  (9, 80) 0.5635760184131379
  (9, 3) 0.1878586728043793
  (9, 81) 0.1878586728043793
  (9, 20) 0.1878586728043793
  (9, 54) 0.1878586728043793
  (9, 117) 0.3757173456087586
  (9, 74) 0.1878586728043793
  (9, 43) 0.1878586728043793
  (9, 38) 0.1878586728043793
  (9, 109) 0.1878586728043793
  (9, 14) 0.1878586728043793
  (9, 0) 0.1878586728043793
==================================================
[[0.         0.         0.         ... 0.         0.25       0.        ]
 [0.         0.         0.         ... 0.         0.         0.        ]
 [0.         0.1934462  0.         ... 0.         0.         0.        ]
 ...
 [0.         0.         0.         ... 0.         0.         0.        ]
 [0.         0.         0.21594036 ... 0.         0.         0.        ]
 [0.18785867 0.         0.         ... 0.         0.         0.        ]]
```

In [162]:

```
df=pd.DataFrame(X.toarray(), columns=tfidf3.get_feature_names())
df
```

Out[162]:

	08002986030	08452810075over18	09061701461	11	12	150	2	2005	21st	3	...	vettam
0	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
1	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
2	0.000000	0.193446	0.00000	0.000000	0.00000	0.000000	0.193446	0.193446	0.193446	0.000000	...	0.000000
3	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
4	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
5	0.000000	0.000000	0.00000	0.000000	0.00000	0.231109	0.000000	0.000000	0.000000	0.231109	...	0.000000
6	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
7	0.000000	0.000000	0.00000	0.000000	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.219673
8	0.000000	0.000000	0.21594	0.000000	0.21594	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000
9	0.187859	0.000000	0.00000	0.187859	0.00000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000

10 rows × 131 columns

Feature Engineering

- Creating new features of transforming existing features using domain knowledge of the data, that makes machine learning algorithm work better.
- Feature Engineering makes Machine Learning Algorithm Learn Better

Creating New Features

1. Length of Documents
    - it may be possible that longer text msgs are more likely to be spam
2. Average Word size within the document
3. Use of punctuation in text

- spam uses too much of punctuation

4. Capitalization of words in document

Transformations

=> Applying some transformations to data can make it work better

1. Power Transformations(x^2, sqrt(x), x^3, x^(a/b), etc)
2. Standardizing data
   - transform skewed distribution to Gaussian Distribution
3. Normalization : bring different features to similar scale
   - convert data of different scale to similar/same scale

Feature Creation

=> Feature creation can be done based on

```
* Message Length
* Punctuation Usage
* Stop Word Usage
* Capitalization Usage
* Average Word Length
```

here we will create features based on

1. Message Length
2. Punctuation Usage

Read Raw Text

In [163]:

```python
import pandas as pd

data=pd.read_csv('SMSSpamCollection', sep='\t', header=None, names=['label','msg'])
data.head()
```

Out[163]:

	label	msg
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives around here though

Create Feature: Message length

In [164]:

```python
data['msg_len']=data['msg'].apply(lambda x: len(x))
data.head()
```

Out[164]:

	label	msg	msg_len
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	111
1	ham	Ok lar... Joking wif u oni...	29
		Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive	

	label	msg	msg_len
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive	155
3	ham	U dun say so early hor... U c already then say...	49
4	ham	Nah I don't think he goes to usf, he lives around here though	61

Create Feature: Punctuation Usage

In [169]:

```
import string
def punctuation_count(txt):
    count=sum([1 for c in txt if c in string.punctuation])
    return 100*count/len(txt)
```

In [171]:

```
data['punctuation_%']=data['msg'].apply(lambda x: punctuation_count(x))
data.head()
```

Out[171]:

	label	msg	msg_len	punctuation_%
0	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there g...	111	8.108108
1	ham	Ok lar... Joking wif u oni...	29	20.689655
2	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive ...	155	3.870968
3	ham	U dun say so early hor... U c already then say...	49	12.244898
4	ham	Nah I don't think he goes to usf, he lives around here though	61	3.278689

Feature Evaluation

Evaluate Created Features

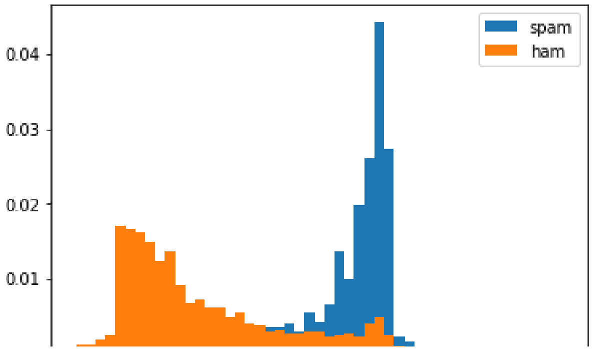
In [172]:

```
from matplotlib import pyplot
import numpy as np
%matplotlib inline
```

Plot msg length for spam and ham

In [178]:

```
bins=np.linspace(0,250, 50)
pyplot.hist(data[data['label']=='spam']['msg_len'], bins, label='spam', normed=True)
pyplot.hist(data[data['label']=='ham']['msg_len'], bins, label='ham', normed=True)
pyplot.legend(loc='upper right')
pyplot.show()
```

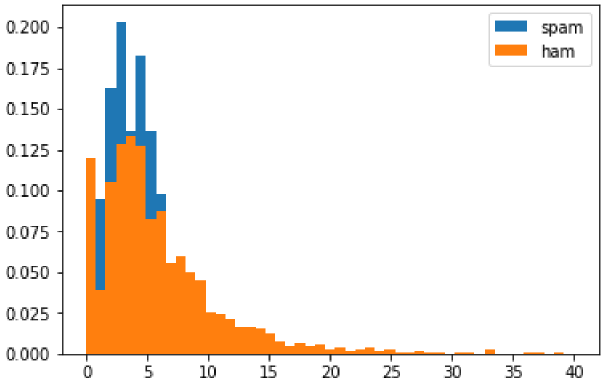


Plot Punctuation_% for spam And ham

In [180]:

```
bins=np.linspace(0,40, 50)
pyplot.hist(data[data['label']=='spam']['punctuation_%'], bins, label='spam', normed=True)
pyplot.hist(data[data['label']=='ham']['punctuation_%'],bins, label='ham', normed=True)
pyplot.legend()
pyplot.show()
```

```
C:\Users\Santosh\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:6521:
MatplotlibDeprecationWarning:
The 'normed' kwarg was deprecated in Matplotlib 2.1 and will be removed in 3.1. Use 'density' inst
ead.
  alternative="'density'", removal="3.1")
```



- Cant conclude much on punctuation as distubution of spam and ham are overlapped

Power Transformations

Transformations

- Changing each data point in a certain column to make the distribution look closer to a normal distribution

```
=> converting right/left skewed data to Normal/ Gaussian distribution
```

Common Transformations

1. Tukey Transformation

- y= X^lambda , lambda>0 -> log(X) , lambda=0 -> -(X)^lambda, lambda<0

2. Box Cox Transformation

- y= (X^lambda -1)/lambda, lambda!=0 -> log(x), lambda=0

Transformation Process

- Determine range of exponents to test
- Apply transformations to each value of the chosen feature
- Determine which transformation yields best distribution, eg. plot histogram and pick which looks closer to a normal distribution

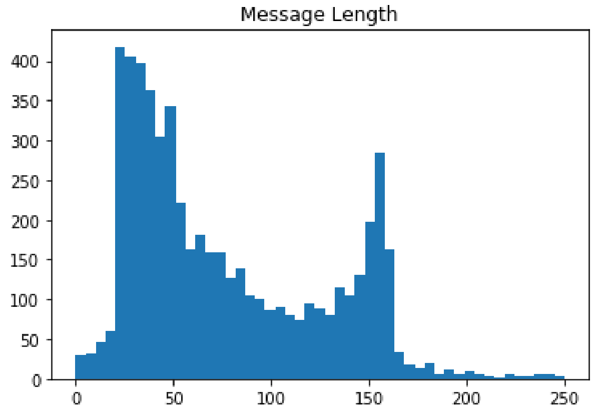
Plot the new Features

In [181]:

```
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

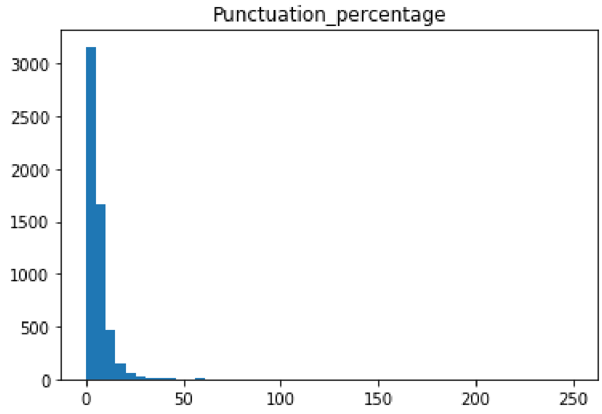
In [184]:

```
bins=np.linspace(0,250, 50)
plt.hist(data['msg_len'], bins)
plt.title('Message Length')
plt.show()
```



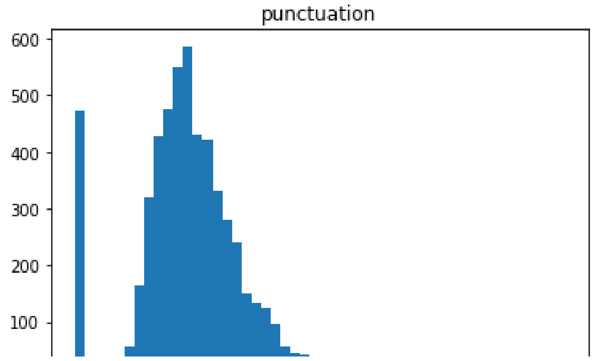
In [185]:

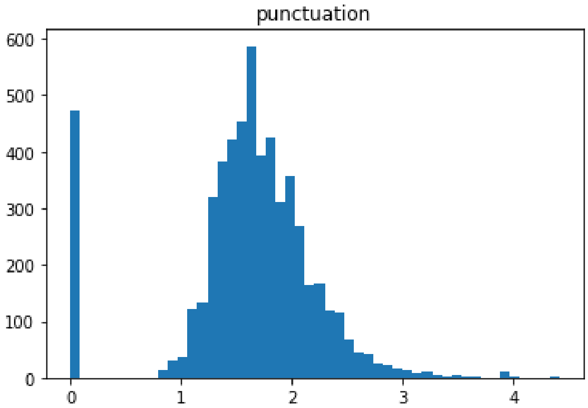
```
bins=np.linspace(0,250, 50)
plt.hist(data['punctuation_%'], bins)
plt.title('Punctuation_percentage')
plt.show()
```

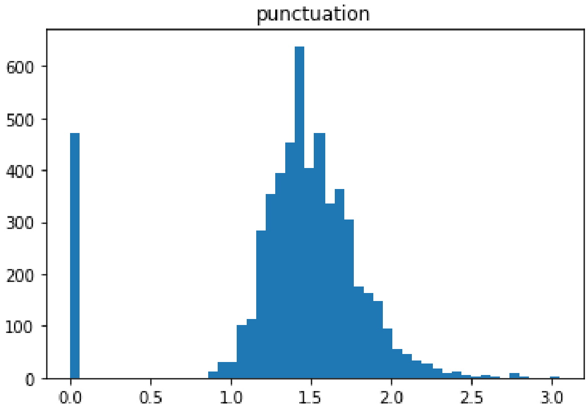


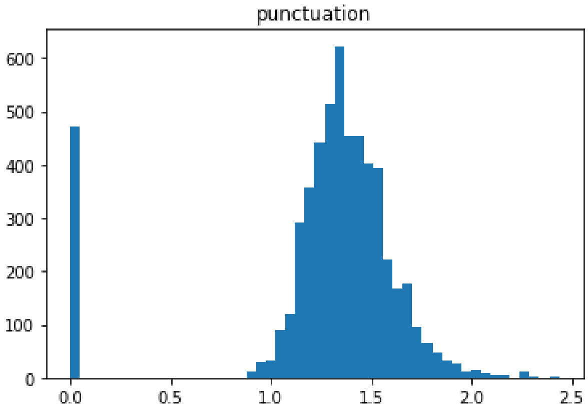
In [190]:

```
for i in [2,3,4,5]:
    plt.hist((data['punctuation_%'])**(1/i), bins=50)
    plt.title('punctuation')
    plt.show()
```









Evaluate the Model : Accuracy, Recall, Precision

```
In [ ]:
```