Result

Question 1

t = 3

Train Error: 0.06444444444444444

Test Error: 0.03875968992248062

t = 7

Train Error: 0.028888888888888888

Test Error: 0.031007751937984496

t = 10

Train Error: 0.015555555555555555

Test Error: 0.03875968992248062

t = 15

Train Error: 0.0

Test Error: 0.023255813953488372

t = 20

Train Error: 0.0

Test Error: 0.023255813953488372

Question 2

The words corresponding to the weak learners chosen in the first 10 rounds of boosting based on the dictionary file are: 'remove', 'language', 'free', 'university', 'money', 'linguistic', 'click', 'fax', 'want', 'de'.

Code

Import Packages and Read Files

In [37]:
```
import numpy as np
import math as ma
```

In [38]:
```
train = open("pa5train.txt", "r")
train = [[int(i) for i in l.strip().split()] for l in train]
```

In [39]:
```
test = open("pa5test.txt", "r")
test = [[int(i) for i in l.strip().split()] for l in test]
```

In [57]:
```
dictionary = open("pa5dictionary.txt", "r")
dictionary = [l.strip() for l in dictionary]
```

Functions

In [40]:
```
def get_h(data, i, sign):
    return [(1 if l[i] else -1) * sign for l in data]
```

In [41]:
```
def get_error(data, h, d):
    return np.dot([h[i] != data[i][-1] for i in range(len(h))], d)
```

In [42]:
```
def get_alpha(error):
    return 1/2 * ma.log((1 - error) / error)
```

In [54]:
```
def get_report(data, boost):
    count = 0
    for l in data:
        sum = 0
        for(alpha, i, sign) in boost:
            h = sign if l[i] else -sign
            sum += alpha * h
        count += 1 if sum * l[-1] < 0 else 0
    return count / len(data)
```

In [55]:
```
def boosting(data, t):
    boost = []
    D = len(data) * [1 / len(data)]

    for e in range(t):
        errors = [(get_error(data, get_h(data, i, s), D), i, s) for i in
range(4003) for s in [1, -1]]
        error, index, sign = min(errors)
        if error >= 0.5: break
        alpha = get_alpha(error)
        h = get_h(data, index, sign)

        temp = []
        for i in range(len(D)):
            temp += [D[i] * ma.exp(data[i][-1] * h[i] * alpha * -1)]
        D = temp

        sum_D = sum(D)
        D = [D[i] / sum_D for i in range(len(D))]
        boost += [(alpha, index, sign)]

    return boost
```

Question 1

In [59]:
```
for t in [3, 4, 7, 10, 15, 20]:
    boost = boosting(train, t)
    train_error = get_report(train, boost)
    test_error = get_report(test, boost)

    print("t =", t)
    print("Train Error: ", train_error)
    print("Test Error: ", test_error)
```

```
t = 3
Train Error:  0.06444444444444444
Test Error:  0.03875968992248062
t = 4
Train Error:  0.051111111111111114
Test Error:  0.03875968992248062
t = 7
Train Error:  0.028888888888888888
Test Error:  0.031007751937984496
t = 10
Train Error:  0.015555555555555555
Test Error:  0.03875968992248062
t = 15
Train Error:  0.0
Test Error:  0.023255813953488372
t = 20
Train Error:  0.0
Test Error:  0.023255813953488372
```

Question 2

In [61]:
```
boost = boosting(train, 10)
words = []
for error,index,sign in boost:
    words += [dictionary[index]]
print(words)
```

```
['remove', 'language', 'free', 'university', 'money', 'linguistic', 'cl
ick', 'fax', 'want', 'de']
```