In [1]:

```
import pandas as pd
from sklearn.model_selection import cross_val_score
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import LogisticRegression

def lower_text(text):
    text_ = text.lower()
    return text_

data_frame = pd.read_csv("linear_train.txt", sep=",", names=['word', 'target'])
data_frame["word"] = data_frame.word.apply(lower_text)
words = data_frame.word
targets = data_frame.target
```

In [2]:

```
transformer = CountVectorizer(min_df=1, ngram_range=(4, 9), analyzer='char_wb', binary=True
matrix = transformer.fit_transform(words)
algo = LogisticRegression(penalty='l2', C=0.5)
algo.fit(matrix, targets)
print cross_val_score(algo, matrix, targets, scoring="roc_auc").mean()
```

```
0.78400263558
```

In [ ]:

```

```

In [3]:

```
test = pd.read_csv('linear_test.txt', names=['word'])
test['word'] = test.word.apply(lower_text)
words = test.word
matrix = transformer.transform(words)
```

In [7]:

```
answer = pd.read_csv('linear_ans_example.txt', sep=',', names=['id', 'answer'], header=0)
predictions = algo.predict_proba(matrix)
answer['answer'] = 1 - predictions
answer = answer.drop(['id'], axis=1)
answer.to_csv('answer.txt', sep=',')
```

In [ ]:

```

```

In [ ]:

```

```