In [ ]:

```
from nltk.tokenize import word_tokenize
from nltk.text import Text
from nltk.util import ngrams
import pickle
```

In [ ]:

```
#function takes a filename as an argument
def preprocess(filename):
    with open(filename, 'r', encoding = 'utf8') as f:
        raw_text = f.read()                                    # read in the text
        raw_text = raw_text.replace('\n', ' ')                 # and remove the newlines

    tokens = word_tokenize(raw_text)                           #tokenize the text

    bigrams = list(ngrams(tokens, 2))                          # use nltk to create a bigrams list

    unigrams = list(ngrams(tokens, 1))                         # use nltk to create a unigrams list

    bigram_dict = {b:bigrams.count(b) for b in set(bigrams)}   #create a bigram dictionary of bigrams and counts

    unigram_dict = {u:unigrams.count(u) for u in set(unigrams)} #create a unigram dictionary of unigrams and counts

    #use the bigram list to create a bigram dictionary of bigrams and counts, [‘token1 token2’] -> count
    countb = 1
    for element in bigram_dict.keys():
        print(element, '->', bigram_dict[element])
        countb += 1
        if countb > 5:
            break

    #use the unigram list to create a unigram dictionary of bigrams and counts, [‘token1 token2’] -> count
    countu = 1
    for element in unigram_dict.keys():
        print(element, '->', unigram_dict[element])
        countu += 1
        if countu > 5:
            break

    return unigram_dict, bigram_dict
```

In [ ]:

```
def main():
    #preprocess the text
    E_Uni, E_Bi = preprocess("LangId.train.English")
    F_Uni, F_Bi = preprocess("LangId.train.French")
    I_Uni, I_Bi = preprocess("LangId.train.Italian")

    #pickle the files
    pickle.dump(E_Uni, open('E_Uni.pickle', 'wb'))
    pickle.dump(E_Bi, open('E_Bi.pickle', 'wb'))

    pickle.dump(F_Uni, open('F_Uni.pickle', 'wb'))
    pickle.dump(F_Bi, open('F_Bi.pickle', 'wb'))

    pickle.dump(I_Uni, open('I_Uni.pickle', 'wb'))
    pickle.dump(I_Bi, open('I_Bi.pickle', 'wb'))
```

In [ ]:

```
if __name__ == "__main__":
    main()
```

In [ ]:

```
```