In [16]:
```
import numpy as np
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.stop_words import ENGLISH_STOP_WORDS



newsgroups_train = fetch_20newsgroups(subset='train', remove=('headers
vectorizer = CountVectorizer(lowercase=True, stop_words=ENGLISH_STOP_W
X_train = vectorizer.fit_transform(newsgroups_train.data)

X_train.toarray()
# X_train.shape #(11314, 101322)
```

Out[16]:
```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]])
```

In [22]:
```
from tqdm import tqdm
def lda(n_dk,n_kw,n_k,z,docs,words,alpha,beta,NITER):
    for tek_iter in tqdm(range(NITER)):
        for i in range(N):
            n_dk[docs[i],z[i]]-=1
            n_kw[z[i],words[i]]-=1
            n_k[z[i]]-=1
            vector_p=np.zeros(K)
            for k in range(K):
                pk=(n_dk[docs[i],k] +alpha[k])*(n_kw[k,words[i]] + bet
                vector_p[k]=pk
            new_z_i=np.random.choice(K,p=vector_p/vector_p.sum())
            z[i]=new_z_i

            n_dk[docs[i],z[i]]+=1
            n_kw[z[i],words[i]]+=1
            n_k[z[i]]+=1
    return  n_dk, n_kw,n_k,z
```

In [20]:
```
M=11314 #amount of texts
W=101322 #amount of different words
N=755809#total amount of words in the corpus
K=20 #amount of tags
NITER=1

alpha=np.ones(K)
beta=np.ones(N)

n_dk=np.zeros(M*K).reshape(M,K) #amount of words in document d assigne
n_kw=np.zeros(K*W).reshape(K,W)
n_k=np.zeros(K) #total amount of words assigned to tag k

X=X_train.toarray()
docs,words=X.nonzero() #print(len(docs)) #it is N=755809
z=[np.random.choice(K) for i in range(N)] #randomly assign tags to wo
for doc, word, cur_z in zip(docs, words, z):
        n_dk[doc, cur_z] += 1
        n_kw[cur_z, word] += 1
        n_k[cur_z] += 1
```

In [*]:
```
n_dk,n_kw,n_k,z=lda(n_dk,n_kw,n_k,z,docs,words,alpha,beta,NITER)
#print(n_kw)
```

```
  0%|          | 0/1 [00:00<?, ?it/s]
```

In [ ]: