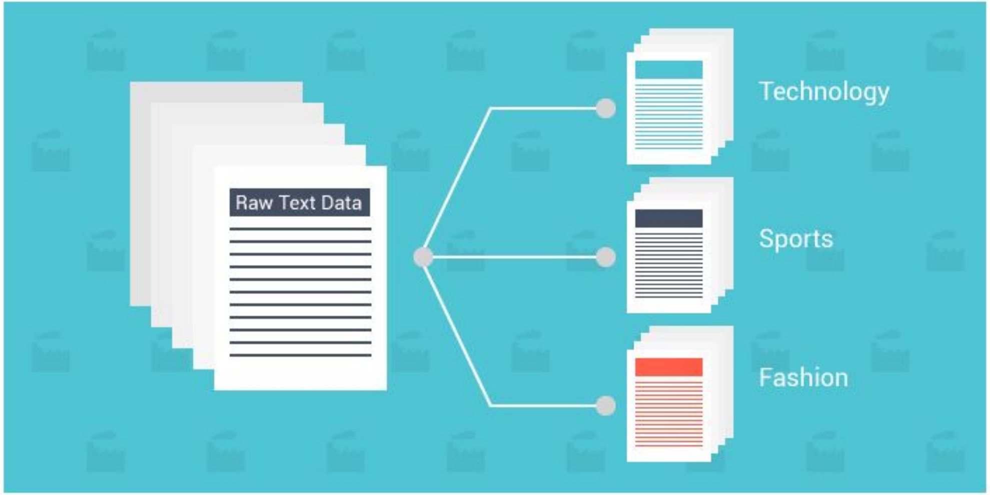
FLOW

1. Project Overview
2. Dataset Overview
3. Feature Extraction and Preprocessing
4. TF-IDF Matrix
5. Word2Vec Model (Embedding)
6. Classifiers
7. SVM and Naive Bayes in brief
8. Results
9. Challenges

What is Text Classification/Document Categorization?

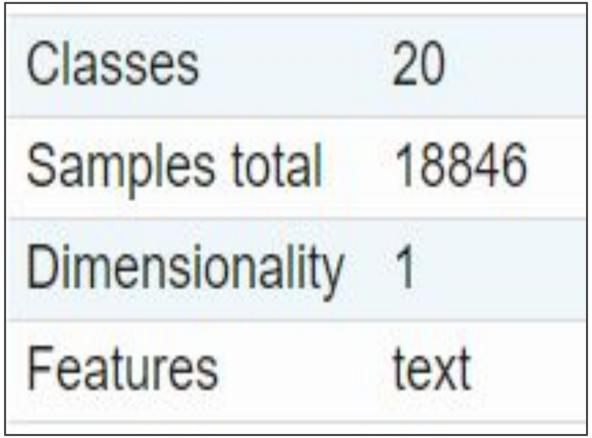
Assign a document one or more classes based on its content.



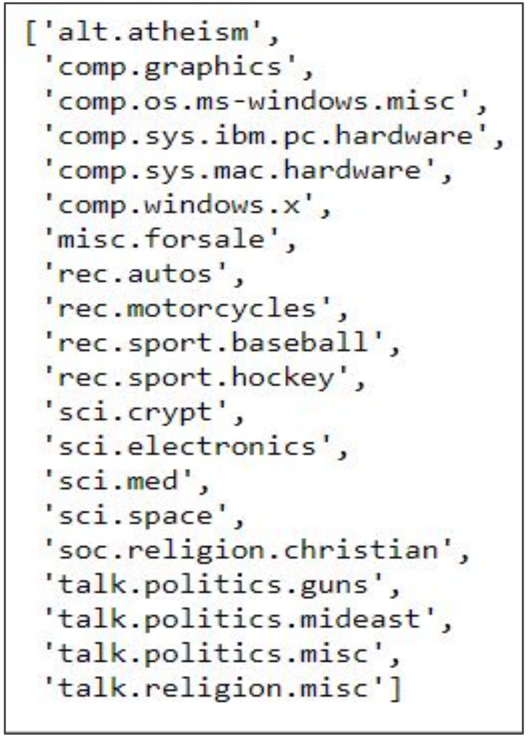
Reference: [3]

Dataset Overview

20 News Group Dataset [2]

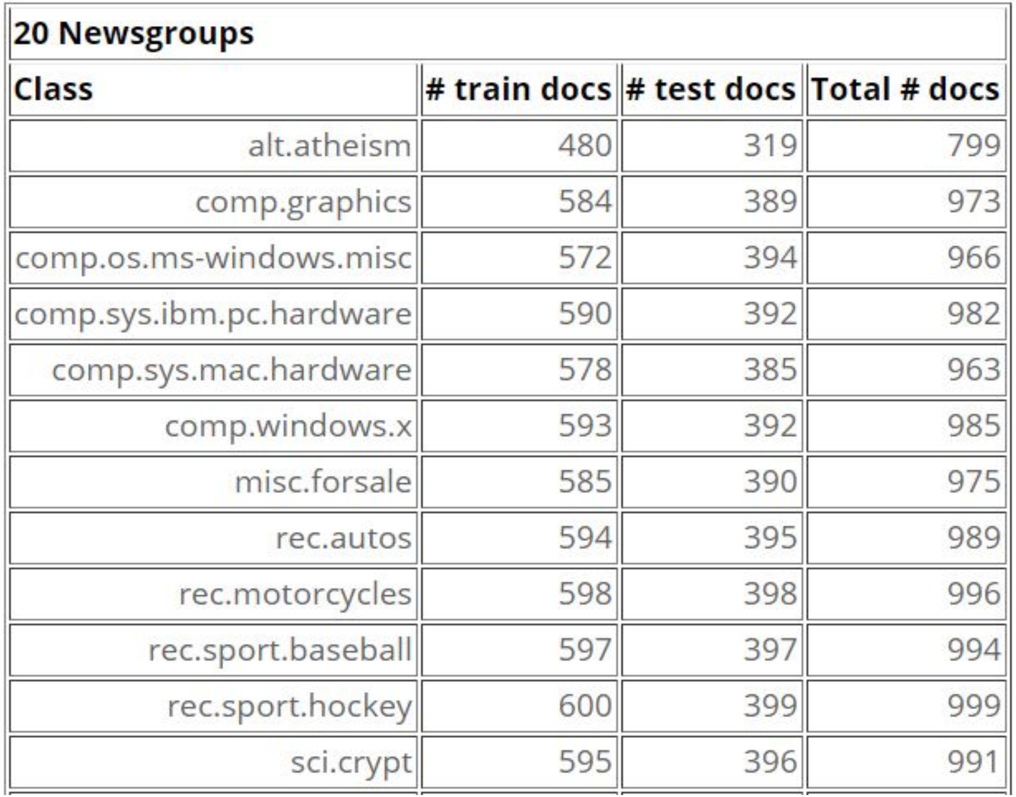


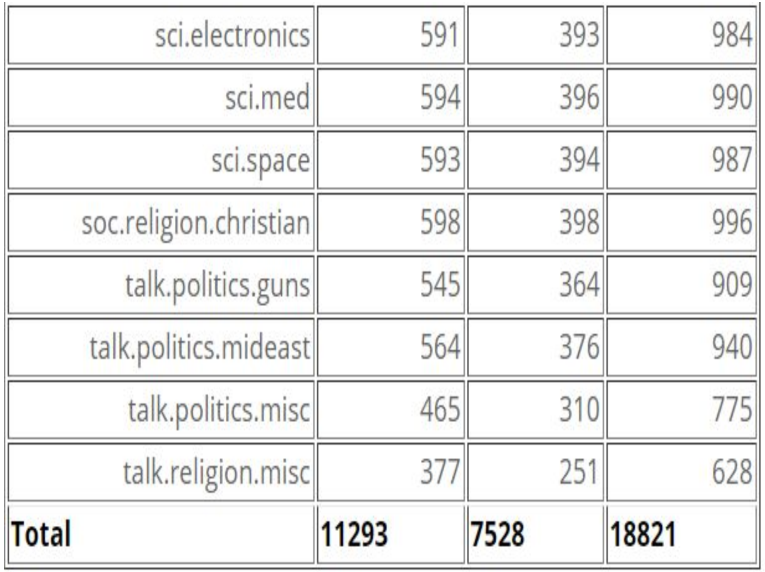
Classes	20
Samples total	18846
Dimensionality	1
Features	text



```
['alt.atheism',
 'comp.graphics',
 'comp.os.ms-windows.misc',
 'comp.sys.ibm.pc.hardware',
 'comp.sys.mac.hardware',
 'comp.windows.x',
 'misc.forsale',
 'rec.autos',
 'rec.motorcycles',
 'rec.sport.baseball',
 'rec.sport.hockey',
 'sci.crypt',
 'sci.electronics',
 'sci.med',
 'sci.space',
 'soc.religion.christian',
 'talk.politics.guns',
 'talk.politics.mideast',
 'talk.politics.misc',
 'talk.religion.misc']
```

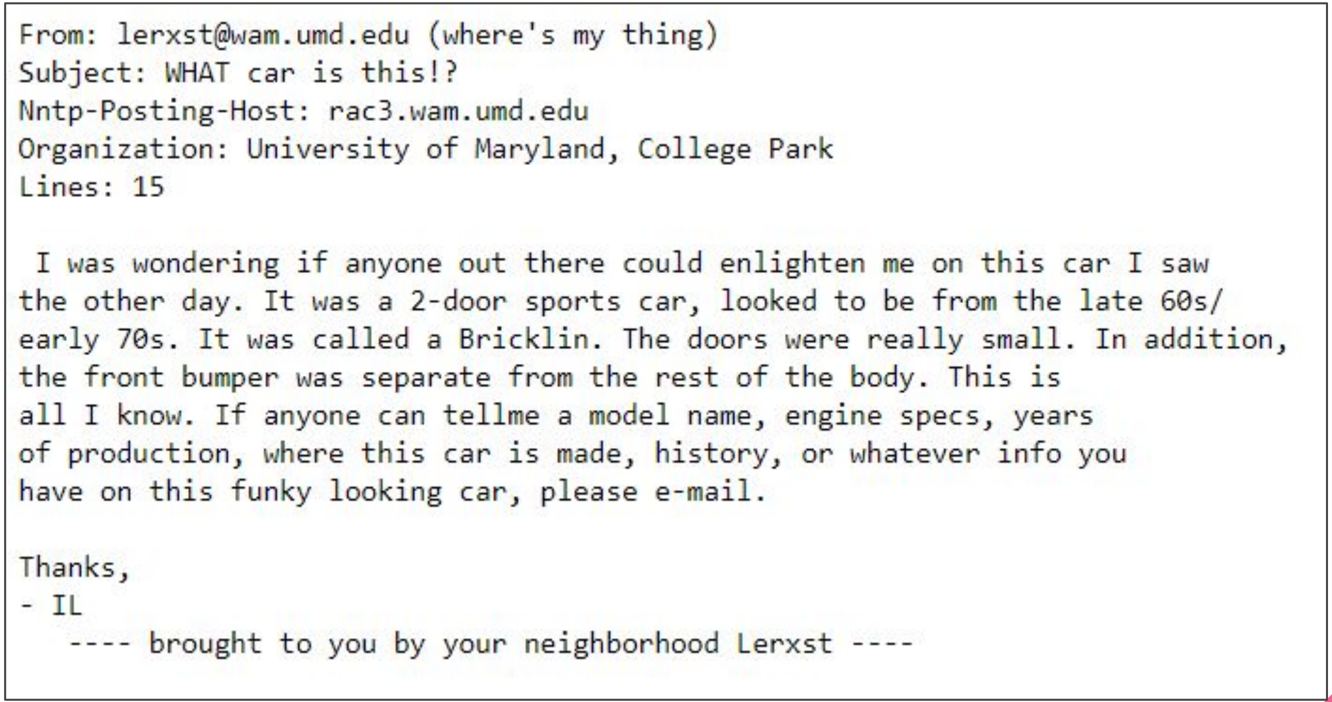
Dataset Statistics: Training vs Testing (60:40) [2]





20 Newsgroups			
Class	# train docs	# test docs	Total # docs
alt.atheism	480	319	799
comp.graphics	584	389	973
comp.os.ms-windows.misc	572	394	966
comp.sys.ibm.pc.hardware	590	392	982
comp.sys.mac.hardware	578	385	963
comp.windows.x	593	392	985
misc.forsale	585	390	975
rec.autos	594	395	989
rec.motorcycles	598	398	996
rec.sport.baseball	597	397	994
rec.sport.hockey	600	399	999
sci.crypt	595	396	991
sci.electronics	591	393	984
sci.med	594	396	990
sci.space	593	394	987
soc.religion.christian	598	398	996
talk.politics.guns	545	364	909
talk.politics.mideast	564	376	940
talk.politics.misc	465	310	775
talk.religion.misc	377	251	628
Total	**11293**	**7528**	**18821**

An Example Document (Class 7 :Misc_for_sale):



```
From: lerxst@wam.umd.edu (where's my thing)
Subject: WHAT car is this!?
Nntp-Posting-Host: rac3.wam.umd.edu
Organization: University of Maryland, College Park
Lines: 15

 I was wondering if anyone out there could enlighten me on this car I saw
the other day. It was a 2-door sports car, looked to be from the late 60s/
early 70s. It was called a Bricklin. The doors were really small. In addition,
the front bumper was separate from the rest of the body. This is
all I know. If anyone can tellme a model name, engine specs, years
of production, where this car is made, history, or whatever info you
have on this funky looking car, please e-mail.

Thanks,
- IL
   ---- brought to you by your neighborhood Lerxst ----
```

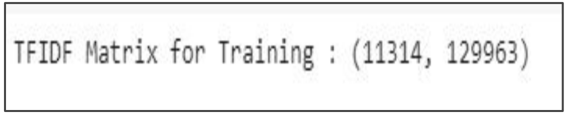
Feature Extraction from Text:

-Stop-words were also removed

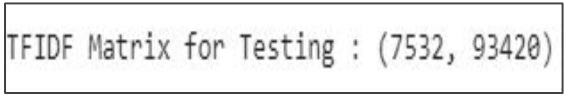
Basically mapping text/string to some real values.

Method 1:

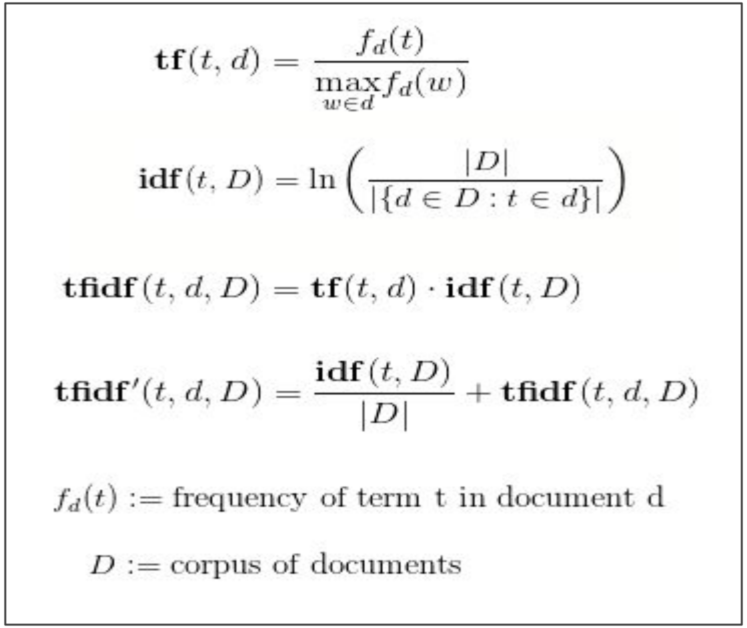
TF-IDF matrix:



```
TFIDF Matrix for Training : (11314, 129963)
```



```
TFIDF Matrix for Testing : (7532, 93420)
```



$$\mathbf{tf}(t,d) = \frac{f_d(t)}{\max\limits_{w \in d} f_d(w)}$$

$$\mathbf{idf}(t,D) = \ln\left(\frac{|D|}{|\{d \in D : t \in d\}|}\right)$$

$$\mathbf{tfidf}(t,d,D) = \mathbf{tf}(t,d) \cdot \mathbf{idf}(t,D)$$

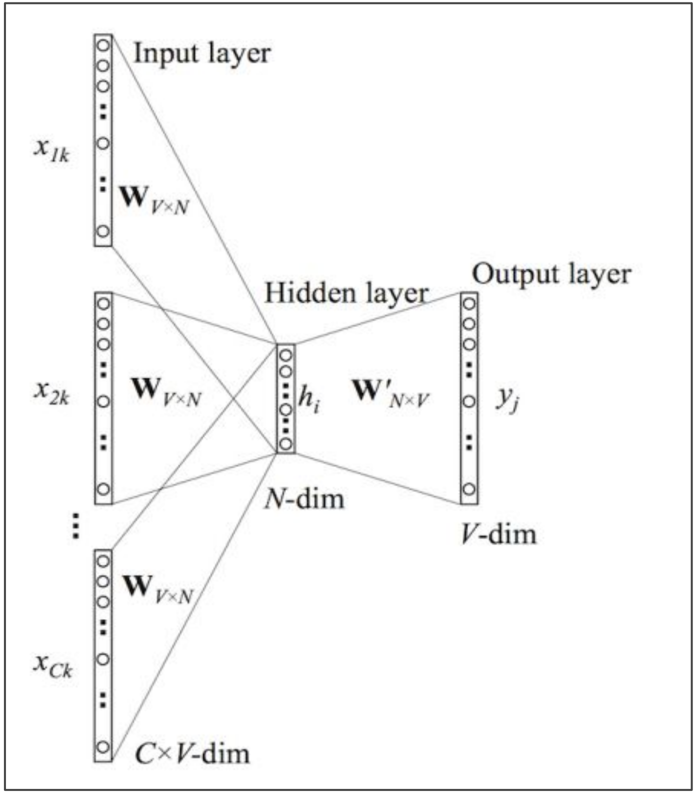
$$\mathbf{tfidf}'(t,d,D) = \frac{\mathbf{idf}(t,D)}{|D|} + \mathbf{tfidf}(t,d,D)$$

$f_d(t) :=$ frequency of term t in document d

$D :=$ corpus of documents

Reference: [4]

Method 2: Word2Vec Model (CBOW) [5]



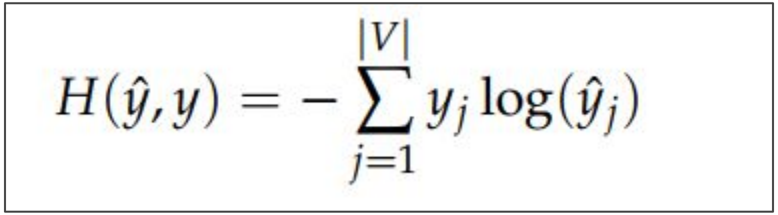
Forward Propagation

$$\mathbf{h} = \frac{1}{C}\mathbf{W}\cdot\left(\sum_{i=1}^{C}\mathbf{x_i}\right)$$

$$u_j = \mathbf{v'_{w_j}}^{T}\cdot\mathbf{h}$$

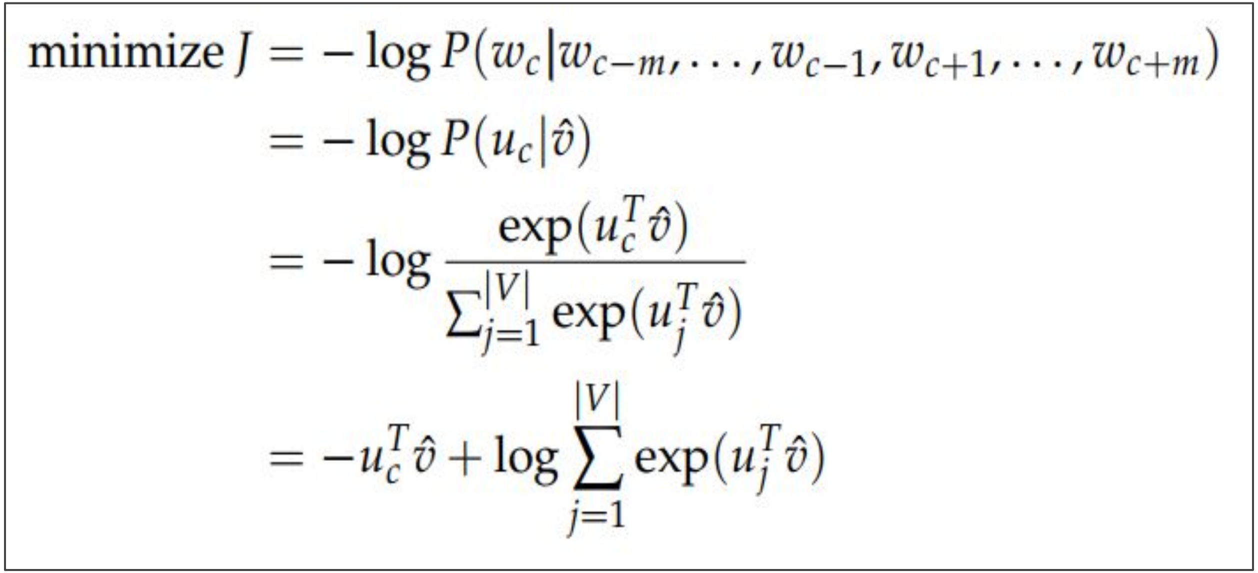
$$y_j = p(w_{y_j}|w_1,\ldots,w_C) = \frac{\exp(u_j)}{\sum_{j'=1}^{V}\exp(u'_j)}$$

Back Propagation in Word2vec



$$H(\hat{y}, y) = -\sum_{j=1}^{|V|} y_j \log(\hat{y}_j)$$

Cross Entropy Loss is calculated and SGD is used to update parameters.

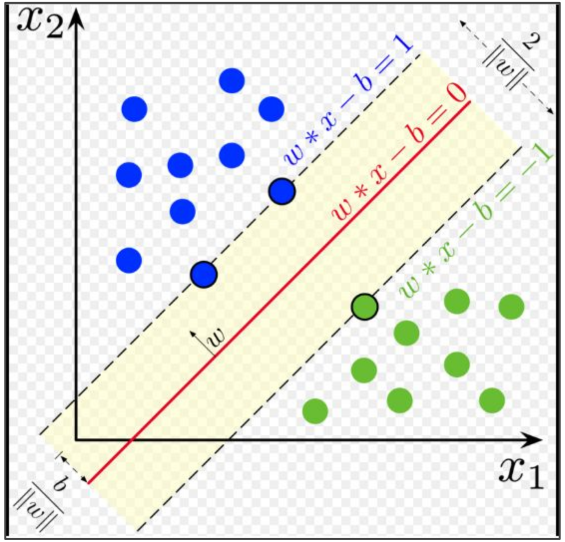


$$\begin{aligned}
\text{minimize } J &= -\log P(w_c | w_{c-m}, \ldots, w_{c-1}, w_{c+1}, \ldots, w_{c+m}) \\
&= -\log P(u_c | \hat{v}) \\
&= -\log \frac{\exp(u_c^T \hat{v})}{\sum_{j=1}^{|V|} \exp(u_j^T \hat{v})} \\
&= -u_c^T \hat{v} + \log \sum_{j=1}^{|V|} \exp(u_j^T \hat{v})
\end{aligned}$$

Classifiers Used:

Support Vector Machine (SVM)

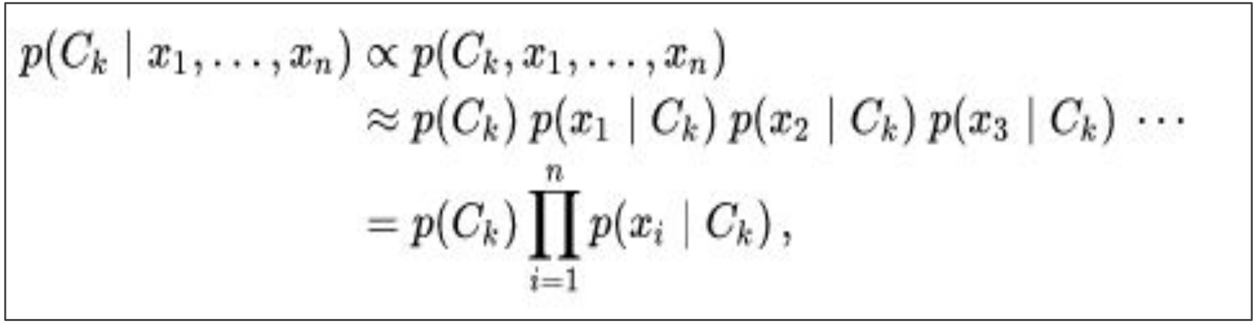
SVM is a widely used maximum margin classifier. It finds a hyperplane separating the classes given to it



Reference : [6]

Naive Bayes Classifier [7]

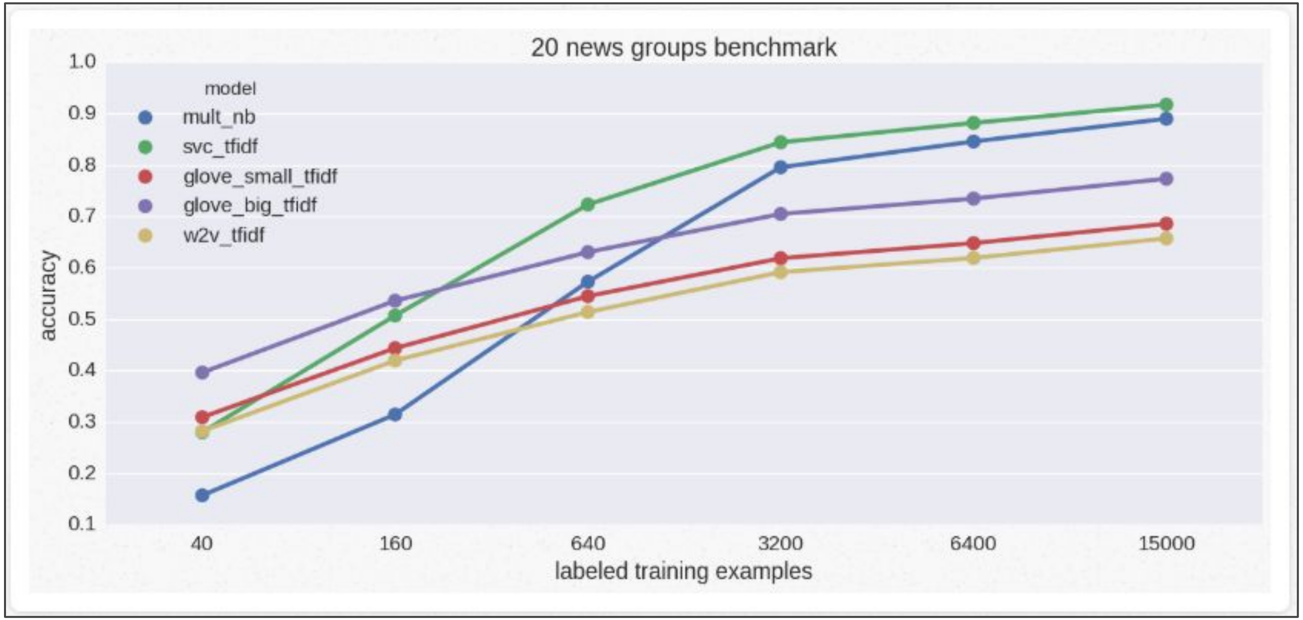
This is a simple probabilistic classifier which uses Bayes' theorem with strong (naive) independence assumptions among the features.



$$\begin{aligned} p(C_k \mid x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n) \\ &\approx p(C_k)\, p(x_1 \mid C_k)\, p(x_2 \mid C_k)\, p(x_3 \mid C_k) \cdots \\ &= p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)\,, \end{aligned}$$

Benchmark Results [8]:

91% Accuracy by Multinomial Naive Bayes on 15000 training examples



Results for TF-IDF + Classifiers

Accuracy on 20 Classes	**Naive Bayes**	**Decision Tree**	**SVM**
TF-IDF + Uncleaned Data	**83.52%**	55.098%	82.38%
TF-IDF + Cleaned Data	83.15%	57.51%	82.12%

Results for Word2Vec + Classifiers

Accuracy	**Random Forest**	**Logistic**	**SVM**
Word2Vec	66.93%	**70.10%**	52.17%

Challenges

1. Tried to implement word2vec from scratch but not optimized for large dataset. Gensim provides optimized word2vec.
2. Multi-class SVM requires linear algebra concepts .
3. Semantics Extraction from data is huge concern.

GITHUB LINK:
https://github.com/rajatgupta1234/Text_Classification

References:

1. Reference Paper: A. Basu, C. Watters, and M. Shepherd. 2003. Support Vector Machines for Text Categorization. In Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03) - Track 4 - Volume 4 (HICSS '03), Vol. 4. IEEE Computer Society, Washington, DC, USA, 103.3.
2. https://www.kaggle.com/crawford/20-newsgroups
3. https://medium.com/@bedigunjit/simple-guide-to-text-classification-nlp-using-svm-and-naive-bayes-with-python-421db3a72d34
4. https://www.joyofdata.de/blog/tf-idf-statistic-keyword-extraction/
5. Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'13), C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Eds.), Vol. 2. Curran Associates Inc., USA, 3111-3119
6. https://upload.wikimedia.org/wikipedia/commons/7/72/SVM_margin.png
7. Rennie, J.; Shih, L.; Teevan, J.; Karger, D. (2003). *Tackling the poor assumptions of Naive Bayes classifiers*). ICML.
8. http://nadbordrozd.github.io/blog/2016/05/20/text-classification-with-word2vec/