Text Clustering and Classification with K-Means and Naive-Bayes

Machine Learning
Petros Mitseas

Goals

- Discover clusters of data based on raw text, with K-Means
- The data are Amazon user reviews, taken from 5 categories:
    - Movies
    - Music Instruments
    - Books
    - Software
    - Clothing
- Take out the labels and let the algorithm discover the clusters

or…

- Use the provided labels to train a Naive Bayes classifier

and finally…

- Package everything into a service that offers:
    - An embedded server for easy deployment
    - Upload and store datasets
    - Train and store K-Means and Naive Bayes models
    - Verify performance using metrics
    - Use the models on new samples
    - A UI tool for performing the above

Feature extraction from text

The TFIDF algorithm is used to transform a document into a numeric vector. Each unique word that appears in the collection of documents (corpus), corresponds to a dimension of the vector. The algorithm combines two quantities:

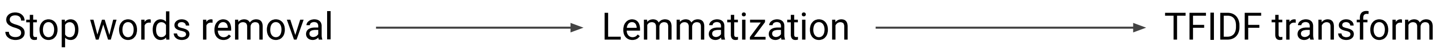
Term Frequency
The number of times a word i appears in the document d divided by the total number of words in that document.

$$TF(i,d) = \frac{f_{i,d}}{\sum_{i' \epsilon d} f_{i',d}}$$

Inverse Document Frequency
Common words that appear in most documents, may not provide useful information. The IDF score of a term is calculated, based on the number of documents containing this term versus the total number of documents:

$$IDF(i,d) = \log \frac{|D|}{|\{d \epsilon D : i \epsilon d\}|}$$

$$TFIDF(i,d) = TF(i,d) \cdot IDF(i,d)$$

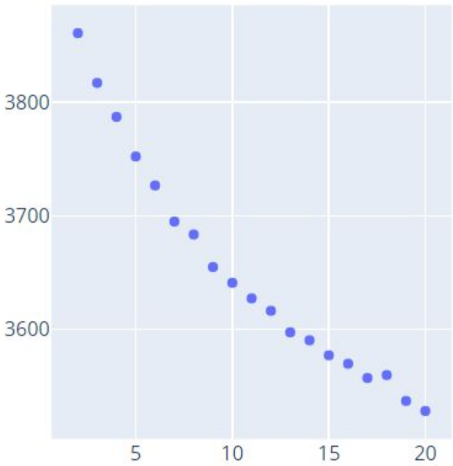


Stop words removal → Lemmatization → TFIDF transform

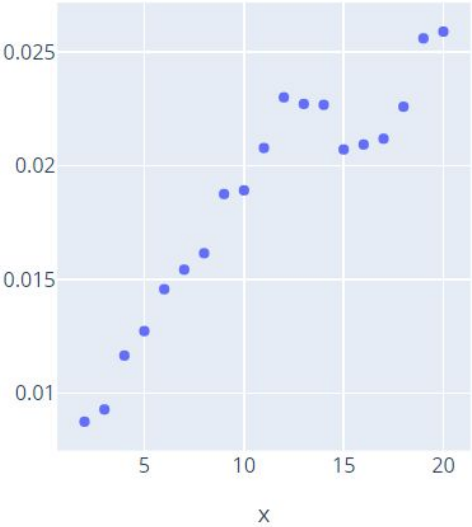
Clustering with K-Means

- Place data points into -fixed number- clusters, based on euclidean distance
- At each iteration assign the data points and recalculate the centroids
- Use silhouette plot and distance from centroids to determine the number of clusters

Mean Silhouette Score

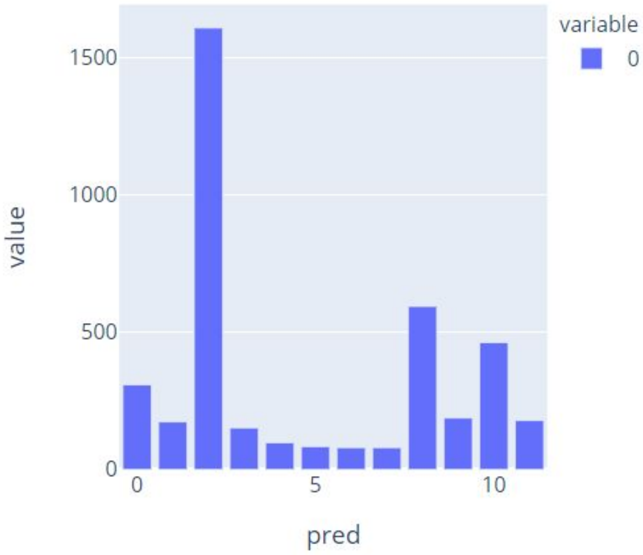


Mean distance from centroids



Clustering with K-Means (II)

Number of points per cluster



- Cluster 2 contains generic points:

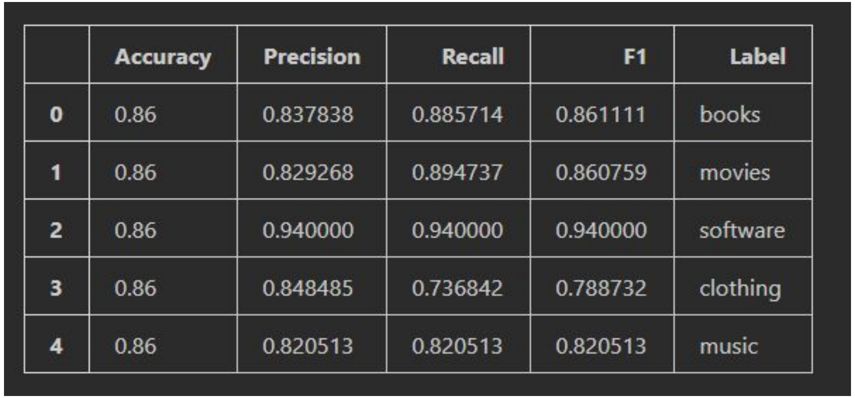
"Im happy plys these aren't fake. Smh yall eeall6 think Converse would let amazom use their name to sell things. Smh. I have a good collection converse. But anywho. Love them." -random guy

- Top terms for most important clusters:

Cluster 0	Cluster 8	Cluster 9	Cluster 10	Cluster 11
movie, watch, good	use, software, program	string, guitar, pick, sound	book, read, child, story	shoe, size, love, wear
I have seen the movie many times and now know how much literally license the movie-makers can take!	*I will update the review if and when I upgrade my system, but for the time being, I didn't expect this incompatibility for a relatively new computer.*	*Excellent quality as expected. Tonal quality is equal to the heavier strings that this new guitar was shipped with*	*After a visit to Delhi, I read this book and greatly enjoyed it -- it added to my visit considerably.*	*A bit large, but that could just be me. Other than that, great quality shoes!*

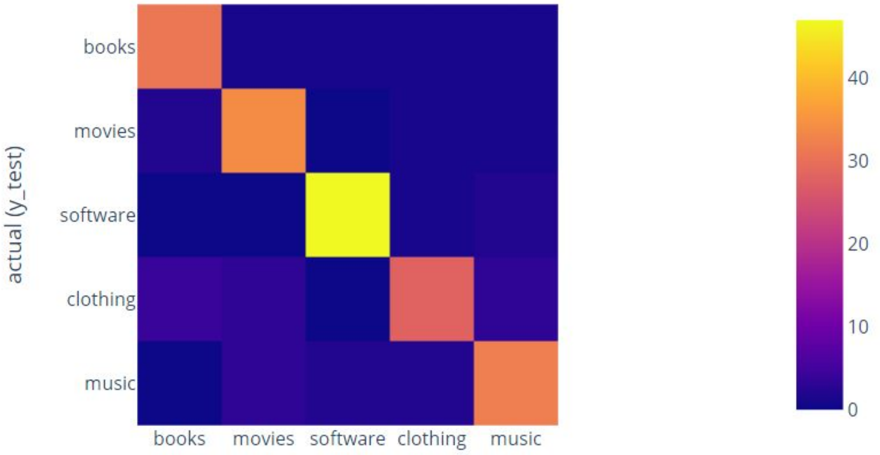
Classification with Naive-Bayes

- Use the labeled data to train a naive bayes classifier
- Assume that the features are individual with one another
- Assume a multinomial probability distribution between input-output



	Accuracy	Precision	Recall	F1	Label
0	0.86	0.837838	0.885714	0.861111	books
1	0.86	0.829268	0.894737	0.860759	movies
2	0.86	0.940000	0.940000	0.940000	software
3	0.86	0.848485	0.736842	0.788732	clothing
4	0.86	0.820513	0.820513	0.820513	music

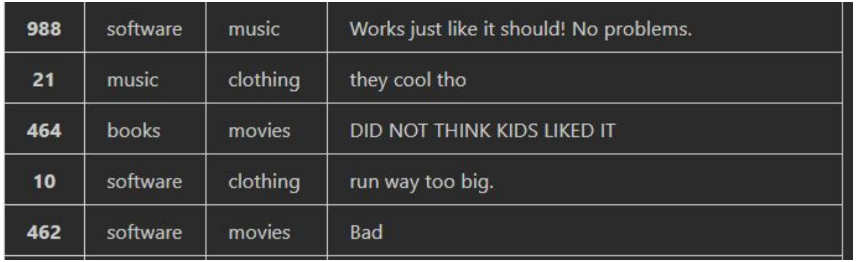
Confusion matrix for dataset with 200 samples per class (80/20 split).



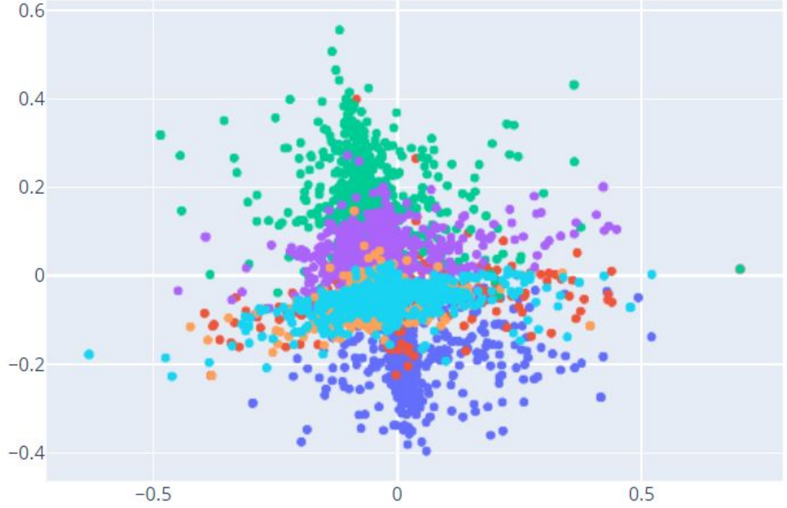
Classification with Naive-Bayes

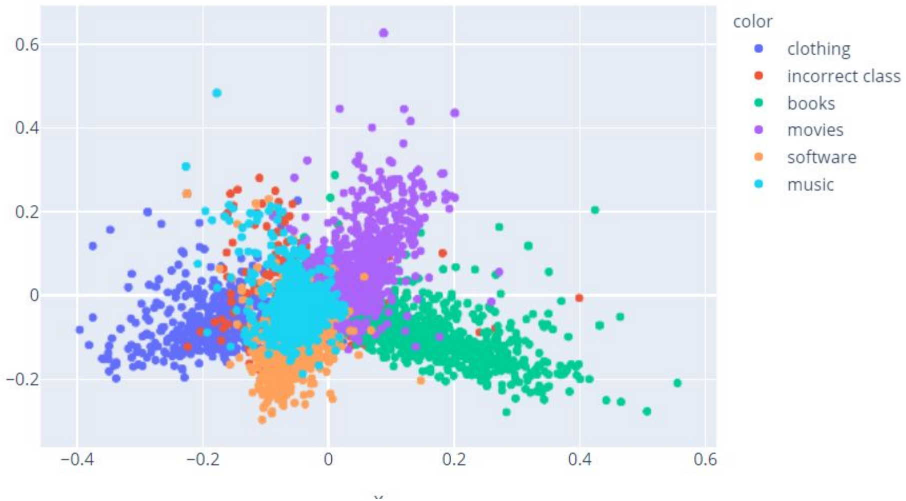
- How to visualize 6011 dimensions?
- PCA transform, then use the 5 most important coordinates

Incorrect labeling examples



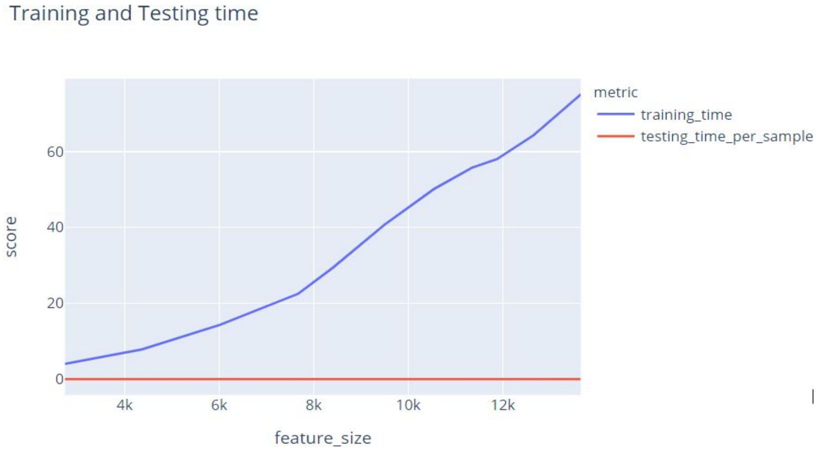
988	software	music	Works just like it should! No problems.
21	music	clothing	they cool tho
464	books	movies	DID NOT THINK KIDS LIKED IT
10	software	clothing	run way too big.
462	software	movies	Bad



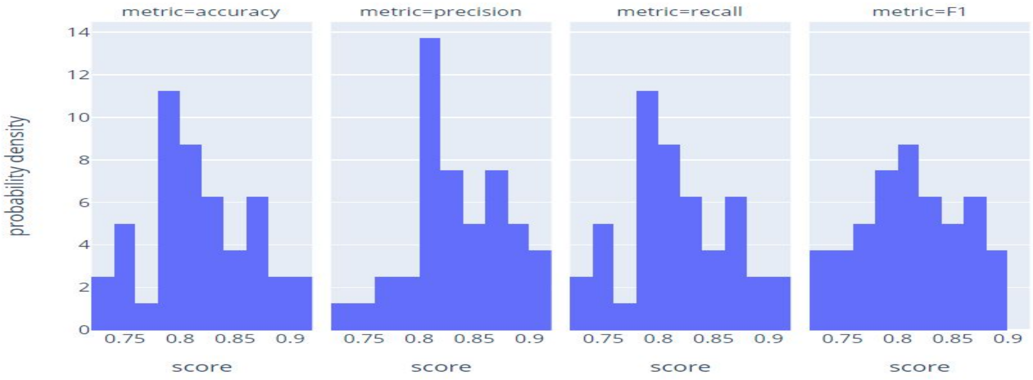


Classification with Naive-Bayes

- Times include TFIDF transform + training time
- TFIDF: O(nlogn), NB: O(n)
- Testing time per sample remains relatively low



- Run the experiment with 200 samples, 40 times
- Capture the performance metrics and plot histograms



Demo

Thank you