Machine Learning and Real-world Data

Example Sheet 1 - Zejia Yang

February 6, 2024

Sentiment Lexicon

1

I exchanged texts with Robert. It was obviously positive. Simple Classifier and Naive Bayes did well on his text.

2

1. Ten examples: no but However Although Despite Though Except Unless Rather Yet Nevertheless.
2. Example sentences:
   (a) By trying to satisfy every kind of viewer, it's possible that SPHERE may end up **pleasing no** one.
   (b) "The Game" is **not a thrilling** roller coaster ride.
3. Adjectives like good, happy, and interesting. Verbs like enjoy, love, and dislike. Adverbs like very much, really, at all. Any words featuring the degree of senitment can have their meaning changed by a *not*.

3

1. Positive Sentiment: Happy, Joyful, Excited, Love, Wonderful
2. Negative Sentiment: Sad, Angry, Disappointed, Frustrated, Hateful
3. Neutral Sentiment: Okay, Normal, Standard, Moderate, Acceptable

We can use a lexicon to classify posts. There are various methods. The most common and simple one is to count the number of positive and negative lexicons and compare them to find out which feelings weigh more in the post.

4

accuracy

$$\frac{328}{412} \times 100\% = 79.61\%$$

5

Accuracy doesn't work well when the classes are unbalanced. The reason is because accuracy measures the proportion of correctly classified instances across all classes, regardless of their imbalance.
In an extreme case for sentiment analysis, the model doesn't learn to predict but just keeps outputting **negative**. If we have a million reviews, 99.99% percent are negative. And this classifier will result in an accuracy of 99.99% for doing nothing.

Naive Bayes

1

a

Estimate conditional probabilities for each class, given each feature:

$$P(A|F1) = \frac{5}{5+5} = \frac{1}{2}$$
$$P(B|F1) = \frac{5}{5+5} = \frac{1}{2}$$
$$P(A|F2) = \frac{0}{0+10} = 0$$
$$P(B|F2) = \frac{10}{0+10} = 1$$
$$P(A|F3) = \frac{3}{3+27} = \frac{1}{10}$$
$$P(B|F3) = \frac{27}{3+27} = \frac{9}{10}$$

b

By applying the Bayes' rules, we have:

$$P(c|F_i) = \frac{P(F_i|c)P(c)}{P(F_i)}$$

Dropping the denominator $P(\sum F_i)$ and $P(c)$ in the numerator, which are both the same for the two class, we have:

$$P(c|F_1, F_3) \propto P(F_1, F_3|c) = P(F_1|c)P(F_3|c)$$

Under the assumption that the probabilities $P(F_i|c)$ are independent given the class c.
For $P(F_i|c)$, we use the training dataset to give a maximum likelihood estimate:

$$P(F_i|c) = \frac{count(F_i, c)}{\sum_{Fi \in \mathbf{F}} count(F_i, c)}$$

Hence, we have the relative probability of A and B.

$$P(A|F_1, F_3) \propto P(F_1|A)P(F_3|A) = \frac{5}{8}\frac{3}{8} = 0.234$$

$$P(B|F_1, F_3) \propto P(F_1|B)P(F_3|B) = \frac{5}{42}\frac{27}{42} = 0.077$$

c

The difference is that we can't simply drop $P(c)$ in the numerator. Hence, $P(c|F_1, F_3) \propto P(F_1|c)P(F_2|c)P(c)$. We have different relative probability of A and B.

$$P(A|F_1, F_3) \propto P(F_1|A)P(F_3|A)P(A) = \frac{5}{8}\frac{3}{8}\frac{25}{100} = 0.0585$$

$$P(B|F_1, F_3) \propto P(F_1|B)P(F_3|B)P(B) = \frac{5}{42}\frac{27}{42}\frac{75}{100} = 0.0574$$

d

F_3 is the most useful for classification in general.
Given the distributions, F_1 occurs the same times in classes A and B, resulting in the same conditional probabilities for both classes. Consequently, F_1 isn't informative.
Conversely, F_2 occurs 10 times in class B and 0 in class A, which means that $F2$ is a strong indicator of class B. However, its absence in class A (or maybe other classes) makes it hard to distinguish classes beyond B.
F_3 appears 27 times in B and 3 times in A, which shows a distinct difference while still being present in A. That suggests F_3 may exhibit varying quantities across different classes, making it a potentially effective indicator for multi-class classification.

e

Filter. Instead of training a model, we can make use of the intrinsic properties of features. Calculate the correlations between features and labels in the dataset for scores like Chi-square, and Pearson correlation coefficient. And then, we select the desired number of features with a high correlation with the labels (targets) but little correlation among themselves.
Wrapper. Method to search the space of all possible subsets of features and also use the dataset in training and testing. We can conduct forward feature selection or backward feature elimination. Having gathered the evaluation scores from the testing data for all combinations of features, we can choose the sets with our preset criteria.
Embedded. Instead of conducting brute-force searches in the feature space, we can incorporate feature interaction into the training process. I would use a random forest model for feature selection. After training a random forest, I can determine the Gini Importance of each feature, which is a useful indicator.

2

It works well because, for document classification or sentiment classification tasks, whether a word occurs or not seems to matter more than its frequency. For example, for a slightly negative word that shows up quite a lot in a text, every occurance has no implication of strong negative feelings (for words like "whatever," it is labelled negative in the task), but duplicate counts may affect NB's prediction by a large amount. Thus, it often improves performance to clip the word counts in each document.
I also tested Binary NB and compared its performance with the original one. Here's the accuracy:

Classifier	**Unsmoothed**	**Smoothed**
Standard NB	0.8	0.805
Binary NB	0.81	0.83

Table 1: Unsmoothed and Smoothed Accuracies for Standard and Binary NB

Statistical properties of language

1

According to Heap's Laws, the relationship between the size of a vocabulary and the size of the text that gave rise to it is:

$$u_n = kn^{\beta}$$

Thus, we will always see new words with a sufficiently large corpus. As for the three words listed: *pferd, abtruce, Kx'a*, There are the answers from ChatGPT:

1. **pferd**: This word is actually German for "horse." While it's not traditionally considered an English word, English speakers with knowledge of German may recognize it. However, for most English speakers who aren't familiar with German, "pferd" wouldn't be recognized as an English word.

2. **abtruce**: This word doesn't appear in standard English dictionaries, and it doesn't seem to have widespread recognition or usage within English-speaking communities. Without a clear definition or context, most people would not consider "abtruce" to be an English word.

3. **Kx' a**: This appears to be a representation of a Khoisan language spoken in Africa. It's not an English word in the traditional sense, but rather a term from a specific language or context. Most English speakers would not recognize or consider "Kx' a" to be an English word