Thumbs up? Sentiment Classification using Machine Learning Techniques - Bo Pang, Lillian Lee, Shivakumar Vaithyanathan

Notes: PENN CIS5300 Jonathon Delemos - Dr. Chris Callison Burch

June 20, 2025

0.1 Abstract

Throughout this paper we will investigate the Naive Bayes, maximum entropy, and support vector machine learning methods. We will discover that the factors that exist in sentiment classifcation make the problem quite challenging.

0.2 Formulas - Naive Bayes

This is the Naive Bayes formula for calculating the probability that document d belongs to class c:

$$P_{\text{NB}}(c \mid d) := \frac{P(c) \cdot \left(\prod_{i=1}^{m} P(f_i \mid c)^{n_i(d)}\right)}{P(d)}$$

Let's define the components:

- $P(c)$: Prior probability of class c
- f_i: The i^{th} word (feature) in the vocabulary
- $n_i(d)$: Number of times feature f_i appears in document d
- $P(f_i \mid c)$: Probability of word f_i given class c
- $\prod_{i=1}^{m}$: Product over all m words in the vocabulary
- $P(d)$: Probability of document d (used to normalize; often ignored in $\arg\max$)

This is a formula we are quite familiar with. We are searching for the probability of class given document. This is the positive or negative sentiment given a particular document. We take the probability of the word raised to the number of times it appears in the document. Then we take that product and reproduce it against every word in the document, also multiplying against the probability of the class and divided by the probability of the document.

0.3 Maximum Entropy

Maximum entropy classification (MaxEnt, or ME, for short) is an alternative technique which has proven effective in a number of natural lan- guage processing applications (Berger et al., 1996). Nigam et al. (1999) show that it sometimes, but not always, outperforms Naive Bayes at standard text classification. It excels by making less assumptions about the data. Its estimate of P (c — d) takes the following exponential form:

$$P_{\text{ME}}(c \mid d) := \frac{1}{Z(d)} exp(\sum_{i} \lambda_{i,c} F_{i,c}(d, c))$$

Let's define the terms.

- $P_{\text{ME}}(c \mid d)$: Probability that document d belongs to class c under the Maximum Entropy model

- $Z(d)$: Partition function or normalization constant, ensures probabilities across all classes sum to 1
- exp: Exponential function e^x
- $\sum_i$: Sum over all features indexed by i
- $\lambda_{i,c}$: Weight parameter associated with feature i and class c, learned during training
- $F_{i,c}(d, c)$: Feature function indicating the presence, count, or strength of feature i in document d for class c

0.4 Support Vector Machines

Support vector machines have been shown to be highly effective. They are large-margin, rather than probabilistic, classifiers, in contrast to Naive Bayes and MaxEnt. In the two-category case, the basic idea behind the training procedure is to find a hyperplane, represented by vector w, that not only separates the document vectors in one class from those in the other, but for which the separation, or margin, is as large as possible. This search corresponds to a con- strained optimization problem; letting $c_j \in 1, -1$ (corresponding to positive and negative) be the correct class of document dj , the solution can be written as

$$\vec{w} := \sum_j \alpha_j c_j \vec{d_j}, \quad \alpha_j \geq 0$$

Let's define the terms.

- α_j - are learned coefficients
- c_j - class model for training model (j+1)
- d_j - feature training model for document j

Big Idea: This is how the weight vector $\vec{w}$ is built.

0.5 Comparison of Formulas

Naive Bayes performs the worst, while **Support Vector Machines** perform the best. It should be noted that bigrams did not greatly improve the efficiency of the algorithms. Furthermore, bigrams rendered algorithms like NB useless, as conditional dependence was required.

0.6 Summary

Throughout this article we compared human classification of sentiment using word selection (60 percent accuracy) with three simple sentiment classification algorithms. We found that the sentiment detection algorithms outperformed the human word selections by rouhgly 20 percent. All trials were more than fifty percent accurate.

0.7 Questions

I've seen the math for both the Maximum Entropy and Support Vector Machines, but how does it work in practice? I'd be intersted to see a problem worked out.