News Classification Using Naïve Bayes and SVM

Yuanhang Zhou[1], Kangle Lyu[2], and Yijie Wang[3]

[1,2]Weiyang College, Tsinghua University
[3]School of Life Sciences, Tsinghua University

January 14, 2024

Abstract

Our project explored using supervised learning algorithms for automatic news classification. News articles from the BBC were collected and processed into individual words, forming a data structure for analysis. Both SVM and Naive Bayes algorithms were trained to categorize articles. Statistical analysis and visualizations revealed key words influencing each category, enhancing model interpretability and understanding which linguistic features determine article classification.

1 Introduction

1.1 Background

The digital era has ushered in a deluge of information, particularly in the form of news readily available via online platforms. Navigating this vast ocean of textual data presents a significant challenge: efficient organization and access.

Text mining, with its ability to extract hidden patterns and insights, emerges as a powerful tool in tackling this challenge. Specifically within text mining, news classification plays a crucial role, simplifying user navigation by categorizing articles into relevant sections. Manual classification relying on human expertise, however, proves costly and time-consuming. This necessitates the need for automation, where supervised machine learning algorithms enable automatic document assignment to one or multiple predetermined categories based on their content.

1.2 Literature Survey

In the field of natural language processing, news text classification has become a relatively mature research area. Early research on news text classification focused on developing statistical methods based on handcrafted features, such as term frequency and inverse document frequency

(TF-IDF). However, these methods were often limited by the quality of the handcrafted features and the availability of training data.

In recent years, there has been a growing interest in using machine learning algorithms for news text classification. These algorithms can automatically learn features from the data, which can lead to improved performance over statistical methods. Some of the most common machine learning algorithms used for news text classification include random forests, support vector machines (SVMs), and deep learning models. Research on news text classification using machine learning algorithms has shown promising results.

1.3 Overview of Our Study

Fueled by a passion for understanding how information is categorized and disseminated, we embarked on a project utilizing established machine learning models. Leveraging Naïve Bayes and SVM methods, we aimed to classify BBC news articles into their five main categories. This hands-on journey with real data collection and analysis served not only to sharpen our data science skills, but also to satiate our curiosity about the inner workings of news classification algorithms and their potential to shape the way we consume information.

2 Methods

2.1 Data Collection

We accessed the official website of BBC and identified five major subcategories for classification: sport, worklife, future, culture, and travel. Data collection was conducted in two stages.

2.1.1 Obtain the URLs of News

Within the Sport category, an index page enumerating all sports subcategories alphabetically was identified. To extract all hyperlinks from the web page, the `rvest` package in R was employed, followed by filtering according to a prescribed format (www.bbc.com/sport/sportname). This process yielded the URLs of all sports subpages.

Subsequently, it was observed that each sports subpage encompasses approximately 20 real-time news reports. Then to procure the URLs of all news items listed on a subpage, a parallel approach was adopted: extraction of all hyperlinks using `rvest` in R and subsequent filtering based on the designated format (www.bbc.com/sport/sportname/[0-9]{8}). By systematically iterating through all subpages, a substantial corpus of news URLs within the Sport category was successfully amassed.

For the remaining four sections, a distinct approach was necessitated due to their comparatively limited number of subcategories. This constraint rendered the previously employed method less conducive to procuring a balanced sample size relative to the Sport category. Notably, a

marked similarity in the structural layout of the main pages across these four sections was observed. Consequently, the Worklife section shall serve as an illustrative exemplar of the URL extraction methodology.

The Worklife section is thematically divided into three distinct segments: How We Work, How We Live, and How We Think. Each static webpage within this section presents approximately 15 articles. Conventional browsing necessitates user interaction with the `Load More Articles` button to initiate the loading of additional articles.

Initially, the employment of `Rselenium` to simulate the browsing process was contemplated; however, the complexities inherent in configuring the environment rendered this approach unfeasible. Therefore, an innovative iterative extraction method was devised. The cornerstone of this method hinges upon the observation that each article invariably links to three additional, thematically related articles. The process unfolds as follows:

1. Extraction of All Primary Article URLs: All URLs on the main page are extracted and filtered according to a predetermined format (www.bbc.com/worklife/article/date-title), yielding a collection of URLs corresponding to primary articles.

2. Iterative Extraction of Related Articles: For each primary article, the URLs of its three interconnected articles are extracted. This iterative process is repeated to incrementally augment the sample size.

3. Elimination of Duplicates: The final stage involves the expurgation of duplicate URLs, culminating in a comprehensive corpus of news URLs pertinent to the Worklife section.

2.1.2 Extract the Primary Textual Content

Preliminary analysis revealed the occurrence of 404 errors for a subset of URLs, potentially attributable to post-publication deletion of news articles. To address this problem, the `httr` package was employed to verify HTTP Response Status Codes. For URLs yielding a successful response (200), the `rvest` package was utilized to extract the textual content within all `p` nodes from the corresponding HTML files, resulting in a vector of paragraphs.

To refine this vector, the `stringr` package was implemented to expurgate punctuation and isolated numerical values, followed by segmentation into individual words. This procedure culminated in the representation of each news article as a vector of words.

It is noteworthy that this extraction methodology may incorporate a modicum of extraneous content, such as copyright information appended to BBC articles or potential timestamps indicating news publication. However, the impact of such residual noise upon subsequent classification endeavors is deemed negligible.

2.2 Support Vector Machine

Support Vector Machines occupy a prominent position within the realm of classical machine learning algorithms, distinguished for their efficacy in data classification tasks. The fundamental principle underpinning SVMs is the identification of a decision hyperplane that effectively partitions the training dataset. Subsequent classification of novel data points is predicated upon their relative positioning with respect to this hyperplane, thereby yielding classification outcomes. SVMs exhibit a pronounced proclivity for adeptly addressing nonlinear classification quandaries and deftly navigating high-dimensional data landscapes.

To ensure optimal performance, meticulous calibration of parameters is paramount. We has devoted considerable attention to the judicious selection of SVM parameters, a discourse that shall be meticulously expounded upon in Results of SVM section.

2.3 Naïve Bayes Classifier

Naïve Bayes classifiers are a family of **linear probabilistic classifiers** which assumes that the features are conditionally independent, given the target class. Despite the apparently oversimplified assumptions, Naïve Bayes classifiers have worked quite well in many complex real-world situations.

Here all words were transformed to lowercase and meaningless words and whitespace fragments containing less than two characters were eliminated. Upon this purification, the distinct words for each article category were concatenated, resulting in five distinct and expansive word vectors.

We then proceeded to perform separate word frequency counts within each category, culminating in the estimation of individual word probabilities for the aggregated dictionary. Employing this approach, we constructed five extensive probability dictionaries, each associated with its corresponding article category. To mitigate potential issues surrounding zero probabilities, we implemented Laplace smoothing with the formula

$$P(w|c) = \frac{n_w^c + 1}{n_c + V} \tag{1}$$

where w is a word, c is an article category, n_w^c is the number of occurrences of word w in category c, n_c is the total number of words in category c, and V is the vocabulary size.

Finally we ventured into the construction of a Naïve Bayes text classifier. The fundamental principle underpinning this algorithm involves calculating the probability of an article's membership within each distinct category, conditional upon the presence of specific words.

The final classification decision emerged by discerning the category associated with the maximum calculated probability, signifying the most probable domain to which the article belongs. To circumvent numerical challenges arising from exceedingly minuscule probabilities, we applied a logarithmic transformation to the equation with the formula

$$\log P(c|x) = \log \frac{P(x|c)P(c)}{P(x)} = \log P(x|c) + \log P(c) - \log P(x) \tag{2}$$

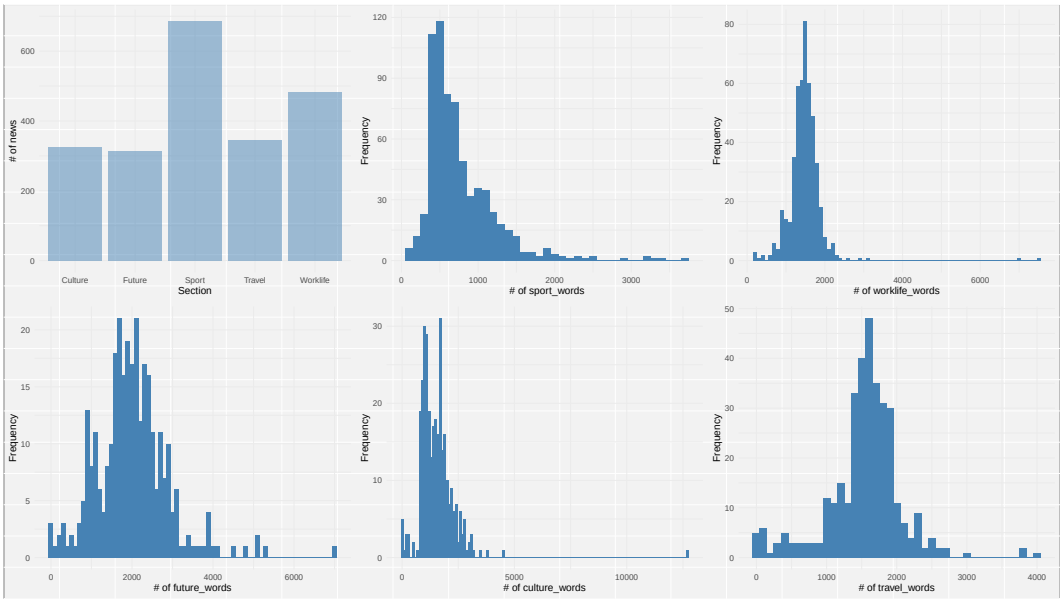


Figure 1: Overview of Data Collection Results

where c is an article category, x is a set of words, $P(x|c)$ is the probability of set x occurring in category c, $P(c)$ is the prior probability of category c, and $P(x)$ is the prior probability of set x.

3 Results

3.1 Overview of Data Collection Results

Figure 1 visually presents the results, showcasing the number of news articles extracted from each section alongside the corresponding word count distribution. The Sport section boasts the highest number of articles (685), followed by Worklife (484), Culture (326), Travel (346), and Future (313).

The word count distribution also reveals interesting trends. While Sport articles tend to be shorter, with most falling within the 0-1000 word range, Worklife and Travel sections exhibit a concentration in the 1000-2000 word range. The Future and Culture sections showcase a broader spread, with articles ranging from 1000 to 3000 words.

3.2 Results of SVM

3.2.1 Construction of Term-Document Matrix

To initiate the exploration of textual patterns within the corpus, we embarked upon the construction of a term-document matrix. This matrix, encompassing 2154 rows and 88005 columns, meticulously mapped the frequency of over 80000 unique words across the expansive collection

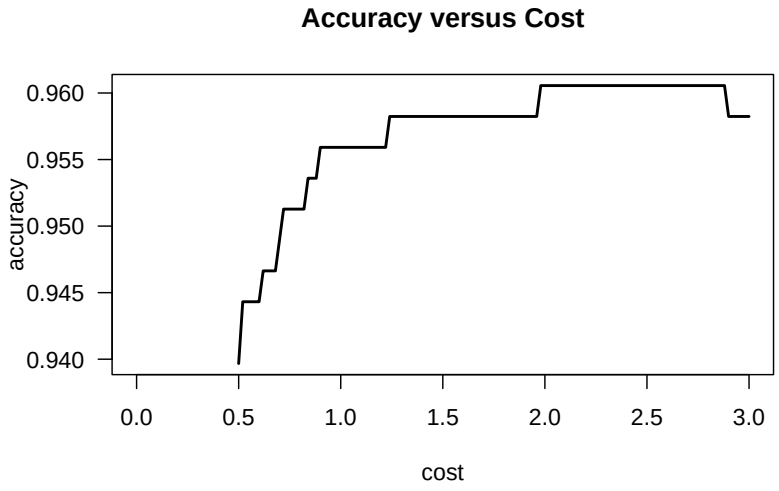


Figure 2: Choice of Cost Parameter

of articles. Each row within this matrix held a numerical representation of the frequency with which each word appeared within a corresponding article.

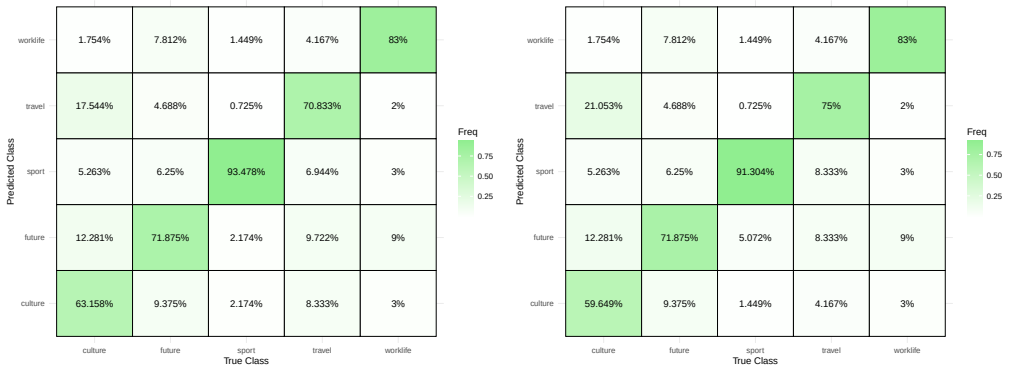
However, recognizing the computational constraints inherent in the `svm()` function within the R environment, we judiciously implemented a dimensionality reduction strategy. By meticulously ranking the frequencies of all words and retaining only the 200 most prevalent terms, we successfully distilled the matrix into a more manageable form, now comprising 2154 rows and 200 columns.

3.2.2 Choice of Cost Parameter

To ensure the rigor and reproducibility of our analyses, we strategically partitioned the text corpus into training and test sets, adhering to a meticulously calibrated 8:2 ratio. Further, we established a steadfast random seed to safeguard the consistency of our results across multiple iterations.

Within the realm of SVM, the `svm()` function harbors a parameter of paramount importance: the cost value, denoted as c. This parameter elegantly dictates the magnitude of penalty imposed upon classification errors in scenarios where linear separability is unattainable. A more generous allocation of c equates to a more stringent penalization of misclassified samples, but an excessively inflated c value can precipitate a phenomenon known as overfitting.

To navigate this delicate balance between model accuracy and generalizability, we embarked upon a meticulous exploration of the optimal c value. By orchestrating a series of SVM models, each meticulously calibrated with a distinct c, we meticulously charted the relationship between cost value and classification accuracy as shown in Figure 2. We elected to embrace $c = 1.98$ to



(a) Confusion Matrix with 200 words (b) Confusion Matrix with 20 words

Figure 3: Results of SVM Classfication

maximize model accuracy and present a confusion matrix as shown in Figure 3a.

3.2.3 Comparison between Models with Different Words

Our exploration of dimensionality reduction through word frequency selection demonstrated promising results. By retaining the top 200 most frequent words, we achieved an overall classification accuracy of 96%, with each individual category exceeding 90%. This signifies the efficacy of the model in achieving robust classification.

Figure 3b illustrates the classification performance when solely employing the top 20 words with the highest frequencies. A notable decline in accuracy compared to the 200-word case is evident, particularly within the Culture category. Its significantly lower accuracy and propensity for confusion with the Travel category suggest that their thematic overlap poses a challenge for the model. In practical terms, this alignment of certain topics between Culture and Travel could explain the misclassifications.

Through further experimentation, we discovered that selecting the top 60 words based on frequency still yielded an impressive accuracy of 90%. This finding suggests a potential trade-off between model performance and computational efficiency, as a smaller feature set offers reduced processing demands while maintaining acceptable accuracy.

3.3 Results of Naïve Bayes Classifier

The confusion matrix of Naïve Bayes classifier is shown in Figure 4. It can be seen that this method has a significant improvement in classification accuracy compared to SVM.

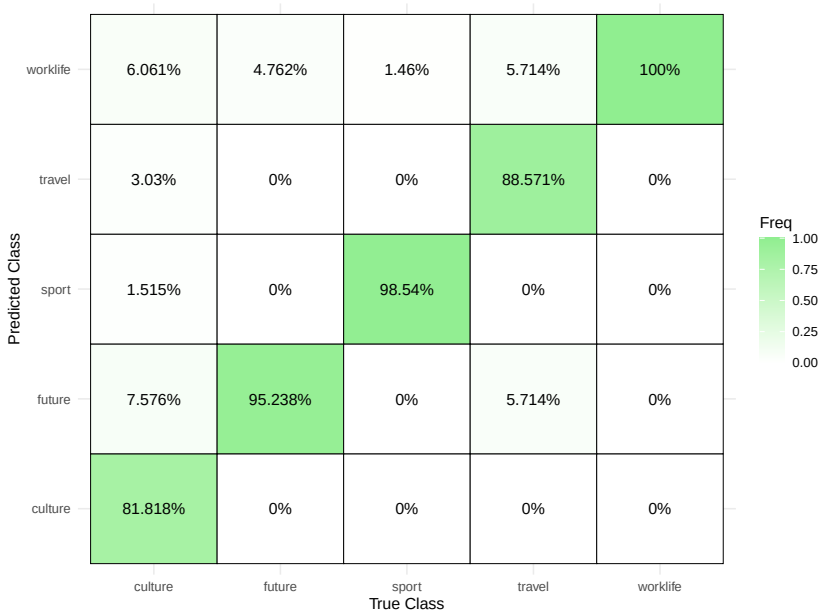


Figure 4: Results of Naïve Bayes Classifier

3.4 Word Frequency Statistics and Analysis

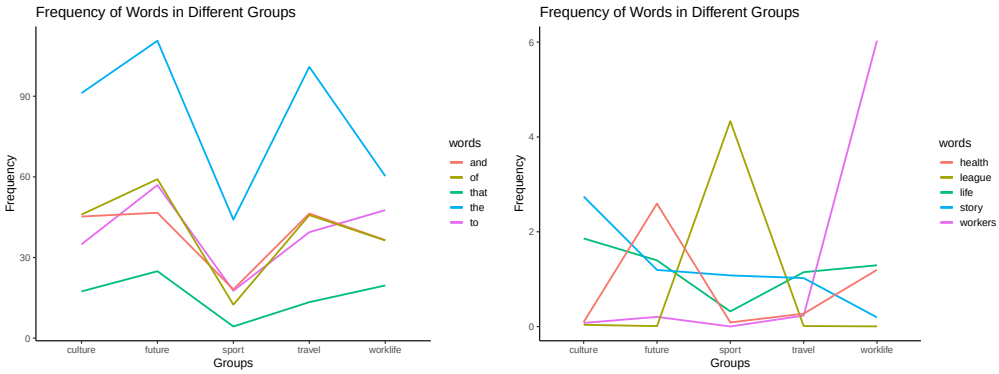
To unveil the most informative words for news text classification, we embarked upon a meticulous exploration of word frequencies within the meticulously constructed term-document matrix.

Figure 5a revealed that the highest-frequency words were predominantly conjunctions, pronouns, prepositions, and articles like and, *it*, *for*, and *the*. These ubiquitous terms, while contributing little to semantic differentiation across categories, served as grammatical scaffolding necessary for textual coherence. Figure 5b unveiled a trove of terms intrinsically linked to specific news topics, such as *health*, *league*, and *workers*.

Notably, the top 50 words were dominated by the aforementioned grammatical elements, exhibiting minimal variation in their frequency across categories. This homogeneity explains the underwhelming performance of the SVM model trained on this limited set. Conversely, words ranked 50-200 in terms of frequency showcased a pronounced association with specific news domains. These thematically relevant terms displayed considerable variance in their distribution across categories, a crucial factor underlying the superior performance of the SVM model trained on the expanded set of 200 words.

4 Discussion

This project focused on the classification of news text. Utilizing web crawlers, we extracted thousands of articles from the BBC website, meticulously processing them into individual words.



(a) Words with Highest Frequencies (b) Specific Words in Each Category

Figure 5: Results of Word Frequency Analysis

A term-document matrix was then constructed, paving the way for classification through both support vector machines (SVM) and Naïve Bayes algorithms.

Further enriching the analysis, we conducted frequency statistics for specific words and visually rendered the results. This insightful visualization sheds light on the underlying dynamics driving the classification, offering valuable interpretability of the models' efficacy.

While this project achieved significant advancements, there is ample room for future exploration. Expanding the research to encompass texts from additional news websites could diversify the dataset and potentially enhance model generalizability. Furthermore, experimenting with diverse classification methods could yield novel insights and potentially further elevate accuracy.

References

[1] Mazhar Iqbal Rana, Shehzad Khalid, and Muhammad Usman Akbar. News classification based on their headlines: A review. In *17th IEEE International Multi Topic Conference 2014*, pages 211–216. IEEE, 2014.

[2] Jeelani Ahmed and Muqeem Ahmed. Online news classification using machine learning techniques. *IIUM Engineering Journal*, 22(2):210–225, 2021.

Team Roles and Responsibilities

Yuanhang Zhou

1. Naïve Bayes Classifier
2. Proposal of Topic

Kangle Lyu

1. SVM Classification
2. Word Frequency Statistics and Analysis
3. Discussion Section of the Essay

Yijie Wang

1. Data Collection
2. Introduction Section of the Essay
3. Integration and Typesetting Work