Applid Machine Learning mid-project

Haomiao Xu
Cornell Tech
hx272@cornell.edu

Thomas Li
Cornell Tech
tl924@cornell.edu

Qiyang Zhang
Cornell Tech
qz392@cornell.edu

Abstract

In recent years, the field of machine learning has seen significant advancements, especially in supervised and unsupervised learning techniques. However, real-world applications often lack complete data annotations, leading to increased interest in semi-supervised learning methods. This project explores semi-supervised learning in the context of sentiment analysis, a classification task that frequently encounters incomplete data labeling due to limitations in data collection or resource-intensive annotation processes.

Our project is divided into two parts: firstly, we aim to leverage unsupervised and hybrid learning techniques to automatically label missing data, thus enriching the dataset. Secondly, we apply supervised learning to this augmented dataset, evaluating whether this preliminary data augmentation enhances classification performance. By integrating semi-supervised approaches, this study intends to provide practical insights into handling missing labels effectively in sentiment analysis tasks, potentially offering robust solutions adaptable to real-world scenarios.

1. Methods

1.1. Data Preprocessing

To standardize the text data, we applied several preprocessing steps:

- **Lowercasing**: All text was converted to lowercase.
- **Special Character and Number Removal**: Non-alphabetic characters and numbers were removed to reduce noise.
- **Lemmatization**: Using the NLTK WordNet Lemmatizer, words were reduced to their base forms, improving consistency.
- **Stopword Removal**: Common English stopwords were removed to focus on meaningful terms.

These steps prepared the text for vectorization, aiming to capture essential information while minimizing noise.

1.2. Vectorization Techniques

We experimented with two vectorization methods to convert text into numerical features:

- **Bag of Words (BOW)**: We used the CountVectorizer with binary=True and min_df=6, creating a sparse representation where each word is a binary indicator.
- **N-Gram**: Using bigrams (N=2), we aimed to capture word pairs, which may provide additional context compared to unigrams.

Each vectorization method provided a unique feature space for the clustering and classification models.

1.3. Semi-Supervised Learning Setup[3]

1.3.1 Clustering for Pseudo-Label Generation

For pseudo-labeling, we applied clustering on the reduced unlabeled feature space using Principal Component Analysis (PCA) to reduce dimensionality to 200 components. Two clustering algorithms were tested:

- **K-Means**: Clusters data points into `K` clusters using the Euclidean distance. We set `n_clusters` to the number of unique classes in the labeled set.
- **Gaussian Mixture Model (GMM)**: A probabilistic clustering approach that assumes data comes from a mixture of Gaussian distributions. GMM was initialized with `diag` covariance and constrained to `K` components.

After clustering, we used the Hungarian algorithm to map clusters to original sentiment labels, producing pseudo-labels for unlabeled data.

1.3.2 Combining Labeled and Pseudo-Labeled Data

We combined labeled and pseudo-labeled datasets, expanding our training data. The combined dataset allowed supervised models to leverage additional information, improving their performance on validation data. [2]

1.4. Supervised Learning Models

With the combined dataset, we trained two supervised classifiers:

- **Multinomial Naive Bayes (MNB)**: A probabilistic model suitable for discrete features, optimized with Laplace smoothing (`alpha=1`).
- **Softmax Regression**: A multinomial logistic regression variant, implemented with the `lbfgs` solver.

Both classifiers were evaluated on validation data, measuring validation accuracy and F1 score to assess their effectiveness in sentiment classification.

2. Results

2.1. Supervised Learning

To evaluate the performance of a purely supervised approach, we trained a Multinomial Naive Bayes (MNB) classifier on the labeled subset of the dataset, excluding any pseudo-labeled data from unsupervised methods. Using a binary Bag of Words vectorization technique with a minimum document frequency of 2, the model achieved an accuracy of 0.9121 and an F1 score of 0.9124 on the validation set. This suggests that the classifier performs well on the labeled data alone, and it provides a strong baseline for comparison against the semi-supervised models.

2.2. Semi-Supervised Learning

The table below shows the validation accuracy and F1 scores for each model combination.

Preprocessing	Clustering	Supervised Model	Validation Accuracy	F1 Score
Bag of Words	K-Means	MNB	0.7587	0.7511
Bag of Words	K-Means	Softmax	0.7469	0.7463
Bag of Words	GMM	MNB	**0.8068**	**0.8058**
Bag of Words	GMM	Softmax	0.6726	0.6732
N-Gram	K-Means	MNB	0.6938	0.7004
N-Gram	K-Means	Softmax	0.6061	0.6224
N-Gram	GMM	MNB	0.7116	0.7081
N-Gram	GMM	Softmax	0.6494	0.6566

Table 1. Validation Accuracy and F1 Scores for each model combination.

2.3. Key Findings

- **Top-Performing Combination**: Bag of Words with GMM clustering and MNB achieved the highest accuracy and F1 score, suggesting that GMM better captures the complex distribution of sentiment data.
- **Impact of Vectorization**: Bag of Words generally performed better than N-Gram, indicating that word pairs did not provide substantial additional context for this task.
- **Clustering Model Comparison**: GMM outperformed K-Means across several configurations, highlighting its suitability for capturing non-linear boundaries in sentiment classes.
- **Why Adding Unsupervised Data Hurt Results**: Surprisingly, the addition of pseudo-labeled data from unsupervised clustering sometimes degraded the model's performance compared to the purely supervised approach. One possible reason is that the clustering process may introduce label noise, especially if the clusters do not align well with the true sentiment categories. Incorrect pseudo-labels can mislead the classifier, causing it to learn suboptimal decision boundaries. Furthermore, when clusters are not well-separated or the clustering method is unable to capture complex sentiment boundaries, the resulting pseudo-labels may be unreliable. This highlights the importance of selecting high-quality pseudo-labels and suggests that in some cases, semi-supervised methods may not always benefit model performance.

3. Discussion

3.1. Insights

3.1.1 Impact of Pre-processing Techniques

Bag of Words vs N-Gram:

Bag of Words generally performs better than N-Gram across different clustering and supervised models, particularly in terms of both Validation Accuracy and F1 Score. Explanation: Bag of Words focuses on individual word frequency, which might be sufficient for capturing sentiment in a simpler and less sparse way. N-Gram, while capturing more context with word sequences, increases complexity and sparsity, which can negatively affect performance unless more sophisticated models are used to manage this complexity. Conclusion: For this sentiment analysis task, Bag of Words provides an effective balance of simplicity and accuracy.

3.1.2 Effect of Clustering Methods

K-Means vs Gaussian Mixture Model (GMM):

GMM generally outperforms K-Means, especially when combined with Bag of Words and the Multinational Naive Bayes (MNB) classifier. Explanation: GMM assumes clusters can have different shapes and be probabilistic, which may better capture the nuances in sentiment data. In contrast, K-Means creates spherical clusters and may not represent sentiment clusters accurately, as sentiments can be more nuanced. Conclusion: GMM is a more suitable clustering method for this dataset, as it likely captures more complex, overlapping clusters in sentiment distribution.

3.1.3 Influence of Supervised Models

Multinomial Naive Bayes (MNB) vs Softmax:

MNB consistently shows higher accuracy and F1 Scores compared to Softmax, regardless of the preprocessing and clustering method used. Explanation: MNB is particularly effective for text classification tasks like sentiment analysis, as it directly utilizes word frequencies. Softmax, while useful, may lack the specific probabilistic handling that MNB has for handling discrete word counts effectively. Conclusion: MNB appears better suited for this task, as it leverages the underlying distribution of word frequencies to improve classification accuracy.

3.1.4 Best Model Combination for Performance

The combination of Bag of Words, GMM, and MNB achieves the highest Validation Accuracy (0.8068) and F1 Score (0.8058), making it the optimal configuration for this dataset. Explanation: This setup leverages the simplicity of Bag of Words, the nuanced clustering of GMM, and the text-classification strength of MNB. Together, they effectively capture the sentiment patterns in the reviews, balancing accuracy and interpret-ability. Conclusion: This configuration is not only the most accurate but also reflects a strategic approach in model selection based on the dataset's characteristics. General Insight Feature Extraction and Model Selection: The results suggest that simpler word-based representations like Bag of Words, when combined with models that are suited for text data (like MNB and GMM), can provide robust performance in sentiment analysis tasks. Using models that handle probabilistic distributions and text frequencies seems to yield the best outcomes, as evidenced by the top scores achieved with this configuration. These insights demonstrate an understanding of how different preprocessing and modeling choices affect performance on this sentiment analysis task. Let me know if you'd like to delve deeper into any specific point! [1]

3.2. Limitations

A limitation of this approach was the absence of deep learning methods, which are often more effective on sentiment analysis tasks but were restricted in this project. Additionally, our use of PCA to reduce feature dimensions may have caused some information loss, impacting clustering quality. Computational constraints limited the exploration of more granular hyperparameter tuning, especially for GMM.

3.3. Future Work

Future improvements could include testing alternative vectorization methods, such as word embeddings (e.g., Word2Vec, GloVe), which may capture semantic nuances better than Bag of Words or N-Gram. Additionally, using self-training or bootstrapping methods could further enhance the quality of pseudo-labels by iteratively refining the model with new labels.

References

[1] T.-H. Hsu, Y.-C. Chang, and H.-H. Chen. Semi-supervised sentiment analysis using heterogeneous text data. *IEEE Access*, 8:158291–158301, 2020.

[2] M. N. Rizve, M. Z. Islam, S. Khan, and N. Barnes. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. In *International Conference on Learning Representations*, 2021.

[3] J. E. Van Engelen and H. H. Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020.