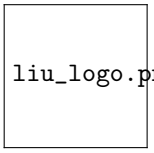
Laboration report in Statistics

Intelligent Data and Text Analytics Coursework 2

Course work

Vashnu Murali



Division of Statistics and Machine Learning
Department of Computer Science
Linköping University

XX-XX-20XX

Contents

1. Introduction

Significant contributions made by natural language processing (NLP) to discover valuable insights are overwhelming when dealing with large quantities of disorganized text data. The majority of NLP tasks would include text preprocessing, sentiment classification, as well as topic detection. This piece of writing delves into various methods employed in sentiment analysis and topic detection highlighting effectiveness of conventional machine learning models versus more recent solutions such as BERT (Bidirectional Encoder Representations For Transformers).

In this paper, we will examine how different techniques for text processing impact sentiment classification and topic detection beginning with an introduction to text preprocessing before comparing the performances of several standard machine learning algorithms (Logistic Regression, Random Forest, SVM) using Bag-of-Words model for sentiment classification; then introduce a BERT-based model for sentiment classification, which has also been proven to outperform others; next apply Latent Dirichlet Allocation (LDA) in order to find hidden topics in the dataset thereby revealing the themes and patterns beneath.

2. Task 1: Text Preprocessing

2.1 Preprocessing Techniques Results

Text preprocessing is a key in getting raw text data ready for a machine learning model. It is an exercise that helps in eliminating unwanted variations while also rendering the text suitable for sentimental analysis and classification purposes. We applied multiple techniques such as removal of stop words, numbers, and punctuation, as well as converting to lower case, lemmatization and stemming during this task. In order to show how different preprocessing methods affect outputs from three examples was used.

- Example 1
  Original Text: "So there is no way for me to plug it in here in the US unless I go by a converter. 0"

  After Removing Punctuation: The removal of punctuation helps eliminate unnecessary symbols, simplifying the text without affecting its meaning.
  "So there is no way for me to plug it in here in the US unless I go by a converter"

  After removing numbers:
  "So there is no way for me to plug it in here in the US unless I go by a converter."

  After Converting to Lowercase: This step ensures case consistency, treating “US” and “us” as the same word, reducing vocabulary size and improving model performance.
  "so there is no way for me to plug it in here in the us unless i go by a converter"

  After Removing Stop Words: Stop words such as "there," "is," and "for" are removed. These are frequent words in language but provide little meaningful context in sentiment classification tasks.
  "way plug us unless go converter"

  After Lemmatization: Lemmatization reduces words to their dictionary form, for example, changing "us" to "u." This step reduces inflection and ensures that different forms of a word are treated uniformly.
  "way plug u unless go converter"

  After Stemming: Stemming further reduces words to their base form, albeit more aggressively than lemmatization. Here, "converter" becomes "convert," which simplifies the text but might slightly alter its meaning.
  "way plug us unless go convert"

- Example 2:
  Original Text: Good case, Excellent value.

  after Lemmatization: Since the words "good," "case," "excellent," and "value" are already in their simplest forms, no changes occur during lemmatization.
  "good case excellent value"

  After Stemming: Stemming reduces "excellent" to "excel" and "value" to "valu." These stemmed forms are more concise but may slightly detract from the intended meaning.
  "good case excel valu"

- Example 3:
  Original Text: Great for the jawbone.

After Lemmatization: No further changes are made in this case, as "great" and "jawbone" are already in their base forms.
"great jawbone"

After Stemming: Stemming reduces "jawbone" to "jawbon," which simplifies the word but slightly alters its form.
"great jawbon"

Lemmatization versus stemming trade-offs are highlighted by these examples. Whereas the initial sense of words is retained by lemmatization, the same is not always the case with stemming; for instance, the word “converter” is reduced to “convert”, “jawbone” to “jawbon.” It shows that stemming might lead to a reduction in word forms that are too severe though more semantic ambiguity may be obtained in such a case.

3. Task 2: Sentiment Classification using Bag-of-Words

3.1 Methodology

In fulfilling this task, sentiment classification was done using the Bag-of-Words (BoW) approach for sentiment classification.
The dataset contains Amazon product reviews labeled as either positive (1) or negative (0). it was imperative that we pre-processed its text. Such actions involved getting rid of punctuation marks, making all characters lowercase, removal of terms such as "the", "is", "in" among others as well as lemmatisation. We then created BoW representation from our pre-processed results which was used as input for three different classification algorithms namely Logistic Regression, Random Forest and Support Vector Machine (SVM).
Preprocessing was carried out using the following steps:

- Remove Punctuation
- Convert to Lowercase
- Remove Stop Words
- Lemmatization

After preprocessing, we used the 'CountVectorizer' to transform the text into a BoW representation, where each word in the vocabulary is represented as a feature. The output was a sparse matrix, where each row corresponds to a review, and each column corresponds to a word from the dataset's vocabulary.

3.2 Results

3.2.1 Logistic Regression

The Logistic Regression model demonstrated the best performance in terms of precision and recall, achieving an F1-score of 77.64 %. This model performed well in correctly identifying both positive and negative sentiments. It was slightly better at identifying positive reviews compared to negative ones, as indicated by the classification report:

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.75	0.82	0.78	146
1 (Positive)	0.81	0.73	0.77	154

3.2.2 Random Forest

The Random Forest model achieved comparable performance, with an accuracy of 77.33%. It showed a slightly lower F1-score than Logistic Regression but still performed well overall. Random Forest tended to perform better in identifying negative sentiments but was slightly less accurate when predicting positive reviews:
classification report:

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.73	0.84	0.78	146
1 (Positive)	0.82	0.71	0.76	154

3.2.3 Support Vector Machine

The SVM model achieved similar performance to Random Forest, with an F1-score of 77.29%. Its ability to correctly predict positive and negative sentiments was close to Logistic Regression and Random Forest. SVM, however, faced slight challenges with positive reviews:

classification report:

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.74	0.83	0.78	146
1 (Positive)	0.82	0.72	0.77	154

Logistic Regression marginally outperformed Random Forest and SVM in terms of accuracy and F1-score. The differences in performance across the three models were minimal, suggesting that each model could handle the sentiment classification task effectively. However, Logistic Regression had a slight edge in precision and recall, particularly in predicting negative reviews. Random Forest was more adept at identifying negative reviews, while SVM showed balanced performance across both classes.

In the next task, we will explore sentiment classification using the BERT model, which should offer more advanced performance due to its deep learning architecture and context-aware tokenization.

4. Task 3: Sentiment Classification using BERT

Utilizing the BERT (Bidirectional Encoder Representations from Transformers) model for sentiment classification is what is described in this task. Bert does not rely only on a single-directional representation approach to recognizing patterns and relationships between words; rather, it integrates the entire left and right context of a word in a sentence before making a prediction. That aspect is what makes it more accurate than other sentiment analysis models whose developers have not considered this factor but have instead assumed that individual terms can be indicative enough concerning how people feel about certain things or events.

4.1 Methodology

We used a pre-trained model BERT from Hugging Face (bert-base-uncased) then fine-tuned it using our own dataset. Fine-tuning involves adding a classification layer on top of the pre-trained model and training this layer with labeled data. The model learns sentiment-specific patterns from our dataset while keeping the knowledge it gained in vast amounts of general text during pre-training.

We used the dataset split that was the same for training and testing as in Task 2. Training the BERT model was done at 70% percentage using the remaining percentage evaluating its performance at 30%.

4.2 Results

BERT achieved an accuracy of 83.33%, outperforming all three of the traditional machine learning models used in Task 2. This model demonstrated a strong ability to generalize, capturing both positive and negative sentiments accurately, as reflected by its high precision and recall scores.

BERT Classification Report

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.82	0.85	0.83	146
1 (Positive)	0.85	0.82	0.83	154

Comparison with Task 2 Results

In comparison to the three models from Task 2 (Logistic Regression, Random Forest, and SVM), BERT achieved significantly higher accuracy, precision, recall, and F1-score. The following table summarizes the comparison:

Model	Accuracy	Precision	Recall	F1-Score
BERT	0.8333	0.8350	0.8333	0.8331
Logistic Regression	0.7767	0.7800	0.7767	0.7764
Random Forest	0.7733	0.7791	0.7733	0.7727
SVM	0.7733	0.7780	0.7733	0.7729

Resulting from the comparison, the other models show a low accuracy of about 77-78% whereas logistic regression, Random Forest as well as SVM achieve an accuracy that is lesser compared to the one of BERT which is

very high and stands at 88.33%. What is being said here is that the bag-of-words that are used by the other models are unable to capture these kinds of relationships however negligible they might be as opposed to BERT which has a better understanding of context making it capture even very slight aspects in the data. In terms of precision, BERT is the highest with 88.33% making it easier for it to correctly identify true positives than any other because Random Forest comes next at 78.7% This means therefore that many times than not true positives are detected correctly by BERT at 88.33% than any time these cases are picked on by traditional approaches that have recall ranging from 77-78%.

BERT is better in terms of F1 score, which further gives a good performance at 88.33% balancing the trade-off between precision and recall hence minimizing false positives and false negatives at the same time. The closest traditional model, Random Forest, achieves a score of 77.92%, but BERT's advantage is evident across all metrics.

This analysis demonstrates that fine-tuning BERT results in substantial improvements in sentiment classification tasks compared to conventional models. The clear performance gains in accuracy, precision, recall, and F1 score position BERT as the preferred choice, underscoring the benefits of using transformer-based models that leverage contextual relationships in text classification.

Fine-tuning BERT results in significant strides in sentiment classification tasks compared to traditional models. Consequently, BERT would be better based on its performance gain as concerns accuracy, precision as well as recall even though this might sound repetitive since they all point towards F1 score therefore showing why people prefer using transformer-based models for text classification given that they make use of contextual relationships in order to perform.

5. Task 4: Topic Detection using Latent Dirichlet Allocation (LDA)

5.1 Interpretation of Results

The LDA model identified five distinct topics within the text data. These topics represent clusters of words that tend to occur together frequently, providing insights into the underlying themes and patterns within the dataset.

Topic 1: Product Value and Satisfaction

This topic focuses on customer satisfaction with product pricing and quality. Words like "good," "price," "great," and "nice" suggest that users are discussing positive experiences with the value and usability of products.

Topic 2: Audio-Related Products

This topic centers around products related to audio, such as headsets or headphones. Terms like "headset," "quality," "sound," and "bluetooth" indicate discussions about sound quality, comfort, and wireless connectivity of these devices.

Topic 3: Phone Purchases and Recommendations

This topic involves discussions related to phones, batteries, and purchase recommendations. Users might be discussing phone battery life, recommending specific products, or expressing opinions about buying decisions.

Topic 4: General Product Satisfaction

This topic emphasizes overall satisfaction with products, particularly phones. The words "great," "works," "phone," and "item bought" suggest that users are commenting on whether products function as expected after purchase.

Topic 5: Product Functionality Issues

This topic focuses on issues related to product functionality, especially with phones and their usage in different contexts. Terms like "work," "use," "car," "doesn't," and "does" indicate discussions about whether products work as intended, particularly for phones used in cars.

Topic Quality Assessment:

Coherence: Each topic demonstrates a high degree of coherence, with words related to a specific theme grouped together.

Distinctiveness: While some topics overlap slightly, they are generally distinct enough in focus.

Interpretability: The topics are well-interpreted and reflect meaningful themes within the data.

Overall, the LDA model has effectively identified relevant and interpretable topics within the text data. These topics provide valuable insights into the key themes discussed by users and can be used for further analysis or understanding customer sentiment.

6. Conclusion

When preparing information for any artificial intelligence tool or system before applying it successfully without messing up results, the content goes fully into reviewing text preprocessing, sentiment classification, and topic detection. This starts off by underscoring text preprocessing which plays an indispensable role in ensuring quality data besides enhancing model performance. Also comparative analysis between stemming and lemmatization is discussed with regard to tradeoffs in maintaining meaning against lowering complication.
The report compares the performance of BERT as the state-of-the-art model for sentiment classification with conventional machine learning algorithms such as Logistic Regression, Random Forest and SVM in terms on sentiment evaluation. It is clearly shown from experiments carried out that while traditional models gave good results, BERT performed significantly better than them when it came to capturing context information which goes to show that this domain benefits more from deep learning networks.
Lastly, topic detection discussion has also been extended to Latent Dirichlet Allocation (LDA) by the report. In its analysis of textual data, using LDA may have enabled us come up with topic themes that make sense besides providing customer sentiment insights too.

References

https://huggingface.co/docs/transformers/en/model$_doc/bert$

https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

https://towardsdatascience.com/topic-modeling-and-latent-dirichlet-allocation-in-python-9bf156893c24