HLT Project: Progress Update

Sentiment Analysis on Amazon Reviews

HLT - Group 11

Angelo Nardone, Riccardo Marcaccio, Matteo Ziboli

May 20, 2024

Dipartimento di Informatica
Università di Pisa

Introduction

HLT Project: Progress Update

- Previous presentation: explored and cleaned the data we worked with.
- Recall that the data represents ”Amazon Reviews.”
- Used this data to address two main tasks, which we'll analyze today.
- Specifically, the tasks we'll cover are:
    1. **Classification (Sentiment Analysis).**
    2. **Negative Reviews Categorization.**

Table of contents

1 Classification

Introduction

HLT Project: Classification

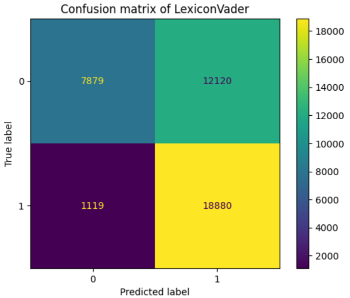
- The goal is to determine whether a review is negative (label=0) or positive (label=1) based only on the title.
- It's a **supervised task** type: we have correct labels to train the model on.
- Used 80% of the data for train and 20% for test.
- Analyze the results obtained using various models, starting from simpler models and progressing to more complex ones.

Lexicon with VADER

HLT Project: Classification

- Used VADER (Valence Aware Dictionary and sEntiment Reasoner) for the first model.
- VADER is a **lexicon** and sentiment analysis tool integrated into NLTK (Natural Language Toolkit).
- Achieved an accuracy of 67%.
- Difficulty in detecting negative reviews.

	precision	recall	f1-score	support
0	0.88	0.39	0.54	19999
1	0.61	0.94	0.74	19999
accuracy			0.67	39998
macro avg	0.74	0.67	0.64	39998
weighted avg	0.74	0.67	0.64	39998

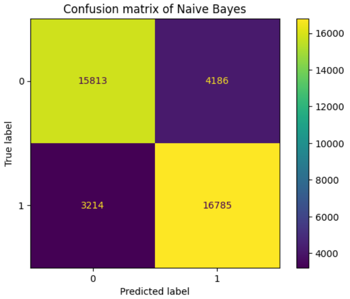


Multinomial Naive Bayes

HLT Project: Classification

- For second classifier, used **Multinomial Naive Bayes**.
- To extract information from textual data, used CountVectorizer and SelectKBest.
- Achieved an accuracy of 81%.
- Much more balance in predicting positive and negative classes.

	precision	recall	f1-score	support
0	0.83	0.79	0.81	19999
1	0.80	0.84	0.82	19999
accuracy			0.81	39998
macro avg	0.82	0.81	0.81	39998
weighted avg	0.82	0.81	0.81	39998

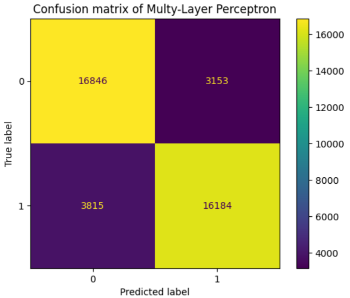


Multi-Layer Perceptron

HLT Project: Classification

- Also used a simple **Multi-Layer Perceptron**, with MLPClassifier from sklearn.
- Same operations for extract information from textual data.
- Achieved an accuracy of 83%.
- Best results among the models seen so far.

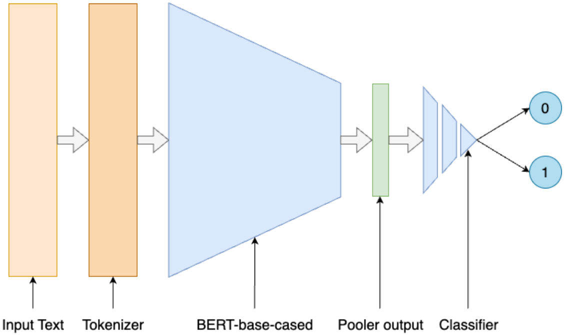
	precision	recall	f1-score	support
0	0.82	0.84	0.83	19999
1	0.84	0.81	0.82	19999
accuracy			0.83	39998
macro avg	0.83	0.83	0.83	39998
weighted avg	0.83	0.83	0.83	39998



LLM using Bert

HLT Project: Classification

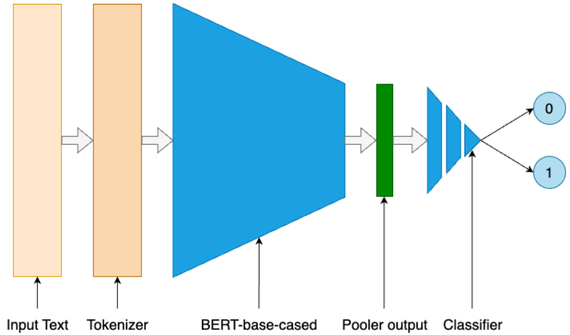
- Final Model: Using a Large Language Model (LLM).
- Used BERT-base-cased to **tokenize** and **generate embeddings** for each sentence.
- A three-layer feedforward neural network is applied for classification.



LLM using Bert

HLT Project: Classification

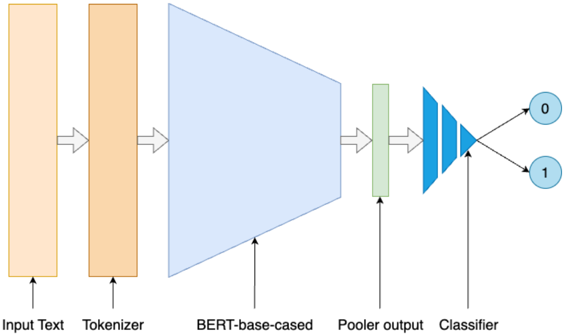
- Used a dual phase for training.
- Initially trained all the parameters: **fine-tuning**.



LLM using Bert

HLT Project: Classification

- Used a dual phase for training.
- Initially trained all the parameters: **fine-tuning**.
- During final epochs, trained only the feedforward classifier, keeping the **Bert weights frozen**.



LLM using Bert

HLT Project: Classification

	precision	recall	f1-score	support
0.0	0.92	0.85	0.88	20000
1.0	0.86	0.93	0.89	20000
accuracy			0.89	40000
macro avg	0.89	0.89	0.89	40000
weighted avg	0.89	0.89	0.89	40000

- For now still few tests with this model, exploring few hyperparameters.
- Achieved an accuracy of 91%.
- Improvement over all previous models.

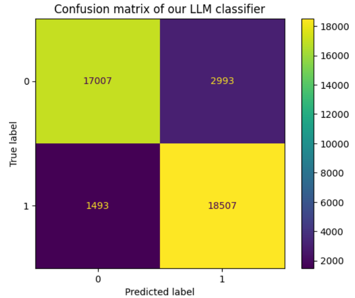


Table of contents

2 Negative Reviews Categorization

Introduction

HLT Project: Negative Reviews Categorization

- This task originates from the following question:

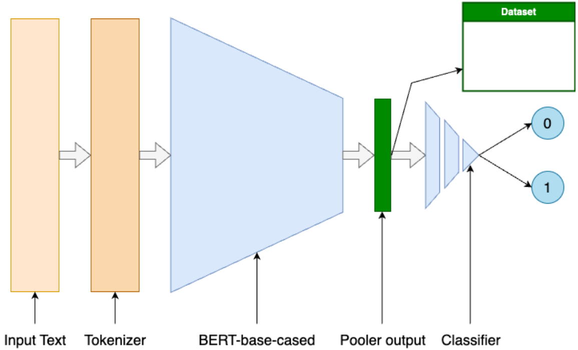
  It is possible determine if a negative review is due to delivery times, potential damage, product quality, or other factors?

- Knowing the answer to this question could be very useful for companies selling products.
- Answer this question by using only the negative reviews from the dataset used so far.
- This is an **unsupervised task**: we do not have labels to associate with the data.

Clustering on Embedding Vectors

HLT Project: Negative Reviews Categorization

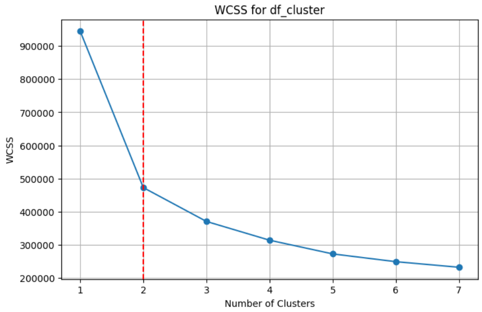
- First step: cluster negative reviews by grouping similar ones together.
- Created a dataset: each title is associated with its **embedding vector** generated by Bert.



Clustering on Embedding Vectors

HLT Project: Negative Reviews Categorization

- Considered only the embedding vectors associated with negative titles.
- Performed clustering on these using K-Means.
- Used the **Elbow Method** to determine the optimal number of clusters.
- We chose 2 as the number of clusters.



Frequent Words

HLT Project: Negative Reviews Categorization

- To capture the main topics of each cluster, counted the most frequent words in them.
- **Tokenized** each title, removing stopwords and punctuation symbols.
- Used methods from NLTK to achieve this.
- Then counted the number of occurrences in each cluster.

Results

HLT Project: Negative Reviews Categorization

- The results obtained were not significant.
- Most of the frequent words were common and not indicative of possible topics.
- Attributed this result to the fact that used only the titles.

First Cluster		Second Cluster	
Word	**Frequency**	**Word**	**Frequency**
book	0,039	time	0,037
bad	0,037	book	0,030
money	0,026	buy	0,028
buy	0,025	waste	0,026
waste	0,023	bad	0,026
great	0,022	great	0,024
poor	0,022	money	0,022

Results

HLT Project: Negative Reviews Categorization

- The results obtained were not significant.
- Most of the frequent words were common and not indicative of possible topics.
- Attributed this result to the fact that used only the titles.

First Cluster		Second Cluster	
Word	Frequency	Word	Frequency
book	0,039	time	0,037
bad	0,037	**book**	0,030
money	0,026	**buy**	0,028
buy	0,025	**waste**	0,026
waste	0,023	**bad**	0,026
great	0,022	**great**	0,024
poor	0,022	**money**	0,022

Embedding Vectors with Word2Vec

HLT Project: Negative Reviews Categorization

- Attempted the same task but on the entire review texts.
- Didn't have the embedding vectors provided by Bert for these texts.
- Generated embeddings for the reviews using Word2Vec.
- For each review, derived an embedding vector by averaging the embedding vectors of all its constituent words (excluding stopwords).

Clustering on Word2Vec Vectors

HLT Project: Negative Reviews Categorization

- Performed clustering on these using K-Means.
- Used the **Elbow Method** to determine the optimal number of clusters.
- Again, 2 was the optimal number of clusters.
- Used the same methods as before to count words frequency in clusters.

Results

HLT Project: Negative Reviews Categorization

- Good Result: there are two clearly distinct topics in the two clusters.
- First cluster: negative review due to the **consumer's personal preferences**.
  - A product is not visually appealing or artistically satisfying.
  - A product did not meet the reviewer's personal taste or expectations regarding content quality.

First Cluster			Second Cluster		
Word	Count	Frequency	Word	Count	Frequency
book	47069	0,0194	product	13243	0,0078
movie	15433	0,0064	use	13101	0,0051
read	13204	0,0054	time	8282	0,0049
story	7901	0,0033	work	8051	0,0048
cd	6927	0,0029	amazon	5096	0,0030
album	6375	0,0026	quality	4602	0,0027
author	5701	0,0024	new	4570	0,0027
music	5530	0,0023	month	3895	0,0023
film	5375	0,0022	problem	3730	0,0022
see	5306	0,0022	old	3523	0,0021
reading	5293	0,0022	receive	3463	0,0020
character	5056	0,0021	return	3422	0,0020

Results

HLT Project: Negative Reviews Categorization

- Good Result: there are two clearly distinct topics in the two clusters.
- Second cluster: many more words suggesting much more **objective issues**.
  - Delivery delays
  - Physical damage or defects in the product.

First Cluster			Second Cluster		
Word	Count	Frequency	Word	Count	Frequency
book	47069	0,0194	product	13243	0,0078
movie	15433	0,0064	use	13101	0,0051
read	13204	0,0054	time	8282	0,0049
story	7901	0,0033	work	8051	0,0048
cd	6927	0,0029	amazon	5096	0,0030
album	6375	0,0026	quality	4602	0,0027
author	5701	0,0024	new	4570	0,0027
music	5530	0,0023	month	3895	0,0023
film	5375	0,0022	problem	3730	0,0022
see	5306	0,0022	old	3523	0,0021
reading	5293	0,0022	receive	3463	0,0020
character	5056	0,0021	return	3422	0,0020

Table of contents

3 Conclusion

Overview

HLT Project: Expanding Model Insights

- Achieved good results for the tasks presented.
- However, our goal by the end of May is to:
  1. Improve the obtained results through further experimentation.
  2. See the results of Negative Reviews Categorization using Bert on whole reviews.
  3. Explore new methods and tasks where there is potential for interesting findings.

Bibliography

[1] Angelido. ”HLT-Sentiment-Analysis: Sentiment Analysis project repository.” *GitHub*. GitHub Repository. (2024)

[2] J. Yosinski, J. Clune, Y. Bengio and H. Lipson. ”How transferable are features in deep neural networks?” In *Advances in neural information processing systems, 27.* (2014)

[3] V. Bonta, N. Kumaresh and N. Janardhan. ”A comprehensive study on lexicon based approaches for sentiment analysis.” In *Asian Journal of Computer Science and Technology 8.S2: 1-6.* (2019)

[4] Hugging Face. Transformers Documentation: BERT. Available on: here. (2024)

[5] PyTorch. PyTorch Documentation. Available on: here.(2024)

HLT Project: Progress Update

Thank you for your attention! :)