IS567 Project - Final Report

Title: Amazon Reviews Product Categorization & Sentiment Analysis

A) Project Summary:

The project aims to classify Amazon product reviews into categories and conduct sentiment analysis for English and German. Goals include implementing multilingual product classification, identifying the optimal model for both classification and sentiment analysis.

B) Descriptive Statistics:

We performed the basic checks like presence of null values and duplicate rows in the dataset. Our dataset does not contain null values or duplicate rows.

Our dataset consists of 31 classes in both languages and exhibits significant category imbalance; for instance, 'home' has 17500+ instances, while 'personal_care_appliances' has around 75 instances in the English language. Similarly, in German, 'home' has 26063 instances, and 'musical_instruments' has 844 instances. Table 1 summarizes the descriptive statistics of the text columns ‘review_body’ for both the languages

Lexical Statistics	English	German
Minimum Words in review	15	16
Maximum Words in review	2270	2915
Average Words in review	105.96	129.77

Table 1. Descriptive statistics of ‘review_body’ column for English and German

C) Pre-processing Steps:

1. Out of the 6 languages, we filtered out two languages of English and German and created separate dataframes of 200,000 rows each.

2. We perform the basic text preprocessing activities on the ‘review_body’ column by removing whitespace, HTML tags, tokenizing and removing stopwords.

3. We initially used lemmatization but found it increased ambiguity, making review classification challenging. For instance, "leggings" for clothing and "legs" for furniture caused confusion. Consequently, we opted not to use stemmers.

4. We noticed standalone numbers in the token list (e.g., '00,' '000,' '2600'). Given our focus on product category classification, we decided to exclude numerical features, processing only word-format numbers (e.g., 'two').

Examples of preprocessed text:

English: *[‘poor quality material fuzzy day one got discolored less month even though kept outdoors covered porch returned item’]*

German: *[‘versand verpackung ordnung produkt spielerei richtig ordentlich nägel manueller maniküre schneller sieht schöner nagelfräser wegschmeißen verschenken’]*

D) Feature Extraction and Feature Selection:

There are numerous approaches for feature extraction including CountVectorizer, TF-IDF, word embeddings, and bag of words. We tried two approaches: CountVectorizer and TF-IDF.

Baseline Model Results using CountVectorizer and TF-IDF vectorizer:

Model	Technique	Number of categories	Precision	Recall	F1 Score
Multinomial Naive Bayes (MNB)	TF-IDF	31	0.486	0.461	0.441
	Count Vectorizer	31	**0.489**	**0.486**	**0.475**

Table 2. Comparison of Count Vectorizer and TF-IDF techniques

Preliminary results from Table 2 show us that count vectorizer performs better than TF-IDF vectorizer, which was the basis of our decision to utilize count vectorizer for further feature selection and model run. Table 3 shows us the metrics for product classification after performing the feature selection using Chi-Square and Mutual Information techniques on 31 classes.

Model Results after k-best Feature Selection:

Type	Model	Precision	Recall	F1 Score
Chi-Square (k = 5000)	Logistic Regression	0.477	0.411	0.426
	MNB	**0.493**	**0.446**	**0.446**
Mutual Information (k = 5000)	Logistic Regression	0.469	0.402	0.416
	MNB	0.491	0.444	0.445

Table 3. Comparison of classification metrics using feature selection across linear models

E) Preliminary Models

(I) ***Product Categorization:***

Baseline model results

To assess classification performance, linear models (Logistic Regression, Naive Bayes, Linear SVC) were chosen as baseline models. We utilized 'weighted' metrics considering class balance, evaluating all 31 categories. Results are summarized in Table 4.

Model	Number of categories	Precision	Recall	F1 Score
Linear SVC	31	0.433	0.445	0.434
Logistic Regression	31	0.458	0.46	0.45
MNB	31	**0.489**	**0.486**	**0.475**

Table 4. Comparison of classification metrics across the baseline linear models

Improvement Methods -

(i) Undersampling and Oversampling to handle imbalanced data:

Addressing the dataset's imbalance (235:1 ratio of majority to minority class), we balanced it using a combination of random undersampling and SMOTE oversampling, aiming for 5000 training instances per class. Linear baseline models were then reevaluated, and classification metrics are presented in Table 5.

Model	Number of categories	Precision	Recall	F1 Score
Linear SVC	31	0.443	0.401	0.406
Logistic Regression	31	0.466	0.407	0.421
MNB	31	**0.469**	**0.455**	**0.442**

Table 5. Comparison of classification metrics across the baseline linear models with undersampling and oversampling

Despite balancing the dataset through a combination of undersampling and oversampling (Table 4), there is no substantial improvement in the model's ability to identify product categories. Notably, the F1 score has decreased for Linear SVC and Logistic Regression models, indicating unsatisfactory performance even with a balanced dataset.

(ii) Filtering dataset based on more than 10000 reviews for product categories

To enhance model performance with 31 classes, we focused on categories exceeding 10,000 reviews. In English, we retained 7 classes (home, apparel, wireless, other, beauty, drugstore, kitchen). For German, 5 classes remained (apparel, home, home_improvement, sports, wireless). Traditional models were trained on this filtered dataset (no undersampling/oversampling), yielding a balanced ratio of 1.7:1 for the majority to minority class. Results are summarized in Table 6 with the best 5000 features.

Reducing the classes from 31 to 7, focusing on those with over 10,000 reviews, significantly improved overall accuracy from 40% to 60%. This refinement allowed traditional methods to better distinguish classes, emphasizing the effectiveness of the revised classification task.

(iii) Exploring complex models

Furthermore, we performed the classification tasks using more complex models such as an Ensemble model (LR + MNB), CNN, LSTM and BERT and observed a significant increase in the performance. Additionally, we decided to remove the category 'other' from the English language as it was creating ambiguity in the review classification. This alone led to a 5% average improvement in performance, as detailed in the error analysis section.

Language	Model	Number of categories	Precision	Recall	F1 Score
English	MNB	6	0.672	0.676	0.672
	CNN	6	**0.703**	**0.68**	**0.69**

	LSTM	6	0.621	0.618	0.62
	BERT	6	0.705	0.705	0.703
	Ensemble Model (LR + MNB)	6	0.680	0.684	0.678
German	MNB	5	**0.69**	**0.683**	**0.673**
	CNN	5	**0.716**	**0.719**	**0.718**
	LSTM	5	0.589	0.592	0.59
	BERT	5	0.694	0.688	0.683
	Ensemble Model (LR + MNB)	5	0.689	0.683	0.674

Table 6. Comparison of classification metrics across all models on categories with more than 10000 reviews for English and German languages

Model Specifications:

All the models use CountVectorizer with 5000 best features.

- Ensemble Model: The pipeline is formed with Multinomial Naive Bayes and Logistic Regression estimators and a soft voting which is used for weighted voting based on probabilities.

- BERT Model: The pretrained model that is used for product classification in English and German was bert-base-uncased and dbmdz/bert-base-german-uncased respectively. The model is BertForSequenceClassifier with AdamW optimizer and CrossEntropyLoss function over 3 epochs. The target variable is label encoded.

- CNN Model: We utilize a Sequential neural model with a one-hot encoded target variable. The model consists of 2 dense layers (64 and 128 neurons), and the output layer uses 'categorical cross entropy' loss, 'Adam' optimizer, and Softmax activation. ReLU activation is applied to the dense layers. Due to resource constraints, the model is trained for 2 epochs.

- LSTM Model: For multiclass classification, we employ a Sequential neural model with a one-hot encoded target variable. The model includes an embedding layer to handle textual data, 1 LSTM layer (64 neurons), and 1 dense layer (128 neurons). The output layer uses 'sparse categorical cross entropy' loss, 'Adam' optimizer, and Softmax activation for multiclass classification. The dense layers utilize ReLU activation. Due to resource constraints, the model is trained for 2 epochs.

(II) ***Sentiment Analysis:***

For sentiment analysis, we used the ‘review_body’ and ‘review_title’ columns as the input features and ‘stars’ as the target variable. We used the same Chi-Square thresholding technique to select the top 5000 features after experimenting with 2000, 2500, and 3000 features.

We performed the 5-class sentiment analysis where the start ratings ranged from 1 to 5 with 1 being the lowest score and 5 being the highest score.

Language	Model	Categories	Accuracy	Precision	Recall	F1 Score
English	MNB	5	0.498	0.485	0.498	0.489
German	MNB	5	0.534	0.525	0.534	0.527

Table 7. Sentiment Analysis on 5 classes using MNB with 5000 best features

Based on the results obtained for Sentiment Analysis on 5 classes in Table 7, we examined the class wise performance and noticed that classes 2,3,4 suffered from lower precision, recall and F1 score values as the model is unable to capture significant complex sentence meaning and context.

So, we decided to perform 3-class sentiment analysis by merging the review classes as:

1, 2 → 'negative' reviews;
3 → 'neutral' reviews;
4, 5 → 'positive' reviews

Language	Model	Categories	Accuracy	Precision	Recall	F1 Score
English	MNB	3	0.668	0.664	0.668	0.666
German	MNB	3	0.709	0.708	0.709	0.709

Table 8. Sentiment Analysis on 3 classes using MNB with 5000 best features

The results obtained for Sentiment Analysis on 3 classes are summarized in Table 8. The 3 class Sentiment Analysis was much more effective than 5 class Sentiment Analysis and there was a 17% improvement in the accuracy.

Upon analyzing class-wise performance, positive and negative classes showed satisfactory results, while 'neutral' reviews were frequently misclassified. To tackle this, we opted to integrate contextual information and semantic relationships into reviews using both custom and pre-trained word embeddings (word2vec and GloVe). Additionally, we experimented with a Convolutional Neural Network model for improved performance.

We employ a Sequential neural model with a one-hot encoded target variable, featuring two dense layers (64 and 128 neurons). The output layer utilizes 'categorical cross entropy' loss, 'Adam' optimizer, and Softmax activation. Dense layers employ ReLU activation. Due to resource constraints, the model is trained for 2 epochs.

Language	Model	Categories	Accuracy	Precision	Recall	F1 Score
English	CNN with word2vec	3	**0.712**	**0.715**	0.712	**0.714**
	CNN with GloVe	3	0.708	0.709	**0.714**	0.712
German	CNN with word2vec	3	**0.74**	0.743	**0.739**	0.74
	CNN with GloVe	3	0.738	**0.745**	0.738	**0.742**

Table 9. Sentiment Analysis on 3 classes using CNN with word embeddings

As we look at the results in Table 9, we see that the CNN with word embeddings had approximately 3-4% improvement in the classification. The 'neutral' reviews saw an improvement in terms of the metrics though the jump was not very high. This suggests that word embeddings can improve classification as they have the power of incorporating meaning in the sentence which is very critical in use cases like Sentiment Analysis.

F) <u>Error Analysis and Insights:</u>

(I) ***Product Categorization:***

We performed error analysis on both top 7 categories and 6 categories after removing the 'other' category. The following are a few interesting examples. Observing 2 examples, which belong to 'other' category, but were misclassified:

1. ['The picture shows the fuel canister with the mosquito repeller attached so I assumed it would be included. However, when I opened the package I discovered there was no fuel. Since it is not included, the picture should be removed. If someone orders this and expects to use it immediately, like I did, they will be very disappointed. Predicted Value is: home']

2. ['We ordered 4 diff brands so we could compare and try them out and this one was by far the worst, once you sat down. You felt squished. I'm 5.5 and 120 lbs. My husband could barely fit. Predicted Value is: apparel']

The first review has been misclassified as 'home', but the actual label is 'other'. On analysis, one can understand that a better suited category for a mosquito repellent would in fact be 'home' rather than 'other'. The second example is related to furniture. This review contains words such as squished, barely fit, etc. This instance is wrongly classified as 'apparel' and the actual label is 'other', whereas it should have been either 'home' or 'furniture' .

Example of a review from 'kitchen' category:

['The graphics were not centered and placed more towards the handle than what the Amazon image shows. Only the person drinking can see the graphics. I will be returning the item. Predicted Value is: other']

When we analyzed the label 'kitchen' as shown above, we found out that one of the reviews was misclassified as 'other' as there was little contextual information in the review, but after human analysis it can be understood that the review should be categorized as 'kitchen'. After removing the 'other' category, affirming our assumption that the category 'other' creates ambiguity, the remaining reviews can be categorized more accurately. The same review was correctly categorized into the label 'kitchen' after removing the 'other' class.

Besides this, there is a general trend of misclassification across labels, where the review is related to failed delivery, or return of the product etc. From the features, it is difficult to understand which category should such reviews belong to as there is no other description of the product in the reviews. Such types of reviews were always categorized as 'home' by the model, and even with better models or even a neural network, it might be difficult to categorize these reviews accurately as seen in the below examples:

1. Actual label is 'drugstore'. Review: *'Amazon never deliver the items. Horrible customer service. Considering new purchase choices. Predicted Value is: home'*
2. Actual label is 'apparel'. Review: "*'ordered product in MARCH has not arrived as of MAY 11. Attempt to contact seller and NO response! Predicted Value is: home"*

3. Actual label is ‘wireless’. Review: *'Nothing like the photo. Boo. Predicted Value is: home'*

This seems to be true in case of German reviews as well:

1. Actual label is ‘apparel’. Review: *“Riesen Probleme mit der Rücksendung. Leider war es nicht das was ich erwartet habe aber der Verkäufer stellte sich erst quer mit der Rücksendung”.* The context is problems with return and it is classified as home

2. Actual label is ‘sports’. Review: *“Achtung Verkäufer erstellt keine Rechnungen auch nach Anfrage nicht. Haben dadurch nur Ärger”.* It is an invoice related issue that is categorized as home too.

The major reason for this misclassification can be the number of reviews in the home category in both languages are the highest, naturally the model is trained accordingly and misclassifies these reviews as home in most cases.

There are reviews which seem to be categorized correctly based on the review body, but have a different actual class label such as:

1. *["When I opened the bag the product smelled of chemical. Which isn't very surprising. Thought I just may have to wash it. I held it up and the fabric was awful. Itchy feeling and very heavy. I tried it on anyway once again I thought after wash this may change. Well it felt heavy on and it has no shape. This is not breathable fabric at all. So I did not wash it repackaged it and sent it back. I bought it to hang around the water park or beach with the kids as a cover up. Nope Predicted Value is: home"]*

2. *['I wish I had taken a picture of the set before I sent it back. But just know that the fork tine split off and the knife had splinters on it that snagged the gauzy bag for storage. The reason I purchased this was to see how the bamboo ones differed. One good thing is that the weight of bamboo is quite a bit less than metal utensils. Couldn\'t comment on "Easy To Clean" since I didn\'t actually use it - afraid of splinters! Predicted Value is: kitchen']* - Although this review should belong to ‘kitchen’, its actual label is ‘home’.

3. *“Wir haben diese Unterhosen zum trainieren gekauft und es ist trotzdem alles daneben gelaufen. Schön und weich sind sie aber zum trainieren fürs trocken werden bringen sie nix, da kann man lieber es anders machen!”.* This review is about training shorts which is categorized into sports but the actual label is apparel. This is ambiguous as can be categorized as both and the model does a fair job.

There might be an overlap between the features, and due to the lack of proper distinction between the product categories, many reviews are misclassified.

(II) <u>Sentiment Analysis:</u>

We performed Error Analysis for the use case on Sentiment Analysis and explored where our classifier was not capable of getting the right results.

A) <u>English:</u>

1. *['So I installed these in my rv and the color seems to very from light to light. Kind of dim and some were just not warm white as described but a blue tint or white color. Predicted Value is: 4']*

The above review has an actual star rating of 2. It does not have many negative words. Additionally the word ‘very’ has been misspelled from ‘vary’. Hence the classifier might have confused it with a strong word exhibiting a positive sentiment. giving it a 4. Hence the review may have been misclassified as a 4 star review by the model.

2. *[‘Can't complain for price, product is a light fragrance not too heavy, great for your travel bag, great thing about this brand is they have several different scents, affordable enough you try them all Predicted Value is: 5’]*

The above review has an actual star rating of 3. The review has a positive tone even though it does not have strong positive terms. There are some negative terms but the meaning is opposite. The overall tone of the review is neither positive nor negative. Even though the actual label is 3, our model is not able to differentiate between the positive and neutral tones. In fact, it does a good job at assigning a better star rating for the review.

B) German:

1. *[Im Glasgefäß befanden sich schwarze Gummikrümel. Das Glasrohr hat scharfe Grate an beiden Seiten. Beide Enden haben kleine Sprünge. Der Gummistopfen riecht unangenehm.Predicted Value is: 4']*

The above sentence when translated to English means: There were black rubber crumbs in the glass jar. The glass tube has sharp ridges on both sides. Both ends have small cracks. The rubber stopper smells unpleasant.

The above review has an actual star rating of 1. The review has a negative tone even though it does not have many negative terms except the word ‘unpleasant’. The model has predicted the star rating of 4 for this review. This is completely opposite the idea and the model is wrong in this case.

2. *[‘Schnelle Lieferung und gute Verpackung aber leider für die Ware kann ich nicht Gleiches sagen. Beim Kochen oder Backen gebildete Kondenswasser bzw Dampf bleib im Ofen wundwasser läuft aus deswegen ich habe das ordnungs zurückgegeben Rückerstattung ziemlich reibungslos gelaufen also Verkäufer 1a aber die Ware nicht in Ordnung ’]*

The above sentence in English means: Fast delivery and good packaging but unfortunately I can't say the same for the goods. Condensation or steam formed when cooking or baking stays in the oven and sore water runs out, which is why I returned it properly. Refund went pretty smoothly, so seller 1a but the goods weren't in order

The above review has an actual star rating of 3. The review has 1-2 negative terms except the word ‘unfortunately’. The review has a negative tone at the start but the refund process was smooth and it adds a positive intonation. So the general emphasis is ‘neutral’. But the model has predicted the star rating of 1 for this review considering the presence of negative terms.

G) Future Scope

We intend to expand our analysis and experiments to include additional languages. Furthermore, we plan to increase the number of training epochs for our model by utilizing a system with increased RAM and a GPU to improve performance. Additionally, our goal is to examine more complex models for the classification of products and sentiment analysis.

References:

Esmaeilzadeh, A., & Taghva, K. (2021). Text classification using neural network language model (NNLM) and Bert: An empirical comparison. *Lecture Notes in Networks and Systems*, 175–189. https://doi.org/10.1007/978-3-030-82199-9_12

Keung, P., Lu, Y., Szarvas, G., & Smith, N. A. (2020). The multilingual Amazon Reviews Corpus. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. https://doi.org/10.18653/v1/2020.emnlp-main.369

Lin, Y.-C., Chen, S.-A., Liu, J.-J., & Lin, C.-J. (2023a). Linear classifier: An often-forgotten baseline for text classification. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. https://doi.org/10.18653/v1/2023.acl-short.160

Sharma, P., & Parwekar, P. (2023). Multiclass classification of online reviews using NLP & Machine Learning for Non-english language. *Intelligent Human Computer Interaction*, 85–94. https://doi.org/10.1007/978-3-031-27199-1_9

- Dataset Link: https://www.kaggle.com/datasets/mexwell/amazon-reviews-multi
- https://nltk.org/
- https://www.nltk.org/api/nltk.html
- https://huggingface.co/dbmdz/bert-base-german-uncased
- https://huggingface.co/transformers/v3.0.2/model_doc/bert.html
- https://scikit-learn.org/
- https://pytorch.org/docs/stable/data.html#torch.utils.data.DataLoader