Explain your opinion - ABSA Analysis

Greta Angolani
University of Milan
gretangolani@gmail.com

October 2024

Abstract

This paper presents an Aspect-Based Sentiment Analysis (ABSA) using the Born Classifier, a text classification algorithm inspired by quantum physics. The objective is to identify candidate aspects from the features extracted by the Born algorithm and assess their contribution to the sentiment polarity of the overall review, with a focus on the sentences where these aspects are present.

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) is a technique in opinion mining that aims to detect and analyze the sentiment expressed towards specific aspects of a text, such as features or attributes of a product, rather than the sentiment of the document as a whole. This allows for a more fine-grained understanding of opinions expressed in user-generated content.

In this paper, we employ the Born Classifier, a text classification algorithm inspired by quantum mechanics, which leverages the notion of superposition of states to classify text documents. The classifier provides a list of features along with their contributions to sentiment polarity (positive or negative).

The goal of this study is to identify candidate aspects from these features and analyze their specific impact on sentiment by focusing only on the sentences where these aspects appear. We aim to explore how sentiment changes when we isolate these aspects and how this influences sentiment classification performance.

This paper is organized as follows: in the section 1 we discuss the dataset and the evaluation methodology adopted; in section 2 we discuss about aspect detection methodology and in section 3 we discuss about aspect detection. In section 4 the results are highlighted and commented and at the end in section 5 and 6 the main related issues and conclusion on the present work are discussed.

2 Dataset and Methodology

The analysis is conducted on a Social Network dataset, composed of 3000 reviews from the AWARE: Dataset for Aspect-Based Sentiment Analysis of Apps Reviews. The dataset contains only two different labels: positive and negative, and is not perfectly balanced between the classes, although the difference is relatively small. For the purpose of this project, the decision was made to under-sample the negative class to match the positive one. After cleaning and preprocessing, the reviews were encoded using the TF-IDF (term frequency–inverse document frequency) vectorizer, a technique that reflects the importance of words in a document relative to a collection of documents or corpus, while the sentiment labels were encoded using LabelEncoder.

Born was then trained and tested, leading to results discussed in Section 3. The main features were subsequently extracted and analyzed to assign them to specific sentences or portions of the text they refer to. A new classification was performed on the subset of sentences, leading to the final prediction results."

3 Aspect Detection

Once we performed classification on the test set, Born returns a list of features extracted from the reviews that were considered for sentiment prediction. Since we aim to extract aspects, which are usually nouns appearing as subjects or objects in the sentences, only those features that have NOUN as part of speech and are classified as subjects, objects, or compound in the reviews were selected as possible candidate aspects. This was accomplished using spaCy, an open-source library for natural language processing in Python. The methodology used for aspects extraction is a syntax-based method.

Moreover, among those features that satisfy the previous criteria, another filter was applied to retain only those with the highest influence on sentiment prediction. For this purpose, the third quartile filter was chosen, resulting in two candidate groups of aspects: positive aspects and negative aspects. Only the features identified as candidate aspects were then assigned to the specific sentences in which they appear, with the constraint that they must be nouns classified as subjects, objects, or compound in the sentence. This process led to a positive dataframe containing all the sentences associated with positive candidate aspects and a negative dataframe for negative candidate aspects.

These are the plots of the most relevant words in positive and negative aspects, illustrating their frequency and impact on sentiment classification.

(a) Positive Candidate Aspects

(b) Negative Candidate Aspects

4 Classification results

In this section we are discussing about the results obtained using Born's Rule for classification. As we can see Born classifier was implemented three times:

1. On the entire dataset, divided in train and test set, for a first classification
2. On the positive aspect dataframe, which contains the sentences to which the positive aspects refer
3. On the negative aspect dataframe, which contains the sentences to which the negative aspects refer

The evaluation metrics used are precision, recall, accuracy, and F1 score (for binary classification problems). These metrics provide a comprehensive evaluation of the model's performance and, by accounting for false positives and negatives, offer a nuanced view of its predictive capabilities.

Accuracy measures the overall correctness of the model's predictions across both positive and negative classes. Precision focuses on the quality of predictions for each class, measuring how many predicted instances for a class (positive or negative) are actually correct. Recall assesses the model's ability to capture all true instances for a class, whether positive or negative. F1 Score provides a balance between Precision and Recall, offering a comprehensive metric that evaluates the model's performance across both classes. The classification results indicate a weak model performance, with an overall accuracy of 50%, meaning the model correctly predicts only half of the time. The precision, recall, and F1 scores are all around 0.50 for both the "positive" and "negative" classes, suggesting that the model's ability to correctly identify each class is close to random guessing.

The F1 score, which balances precision and recall, further reinforces that the model struggles to differentiate between positive and negative sentiment. It does not show significant bias towards either class, but this balance is due to the overall poor performance rather than effective classification.

Overall, these results indicate that the model is currently under performing and would benefit from further tuning or a more robust approach to improve its predictive ability.

	precision	recall	f1-score	support
negative	0.49	0.50	0.50	270
positive	0.50	0.50	0.50	275
accuracy			0.50	545
macro avg	0.50	0.50	0.50	545
weighted avg	0.50	0.50	0.50	545

Figure 2: Evaluation Metrics

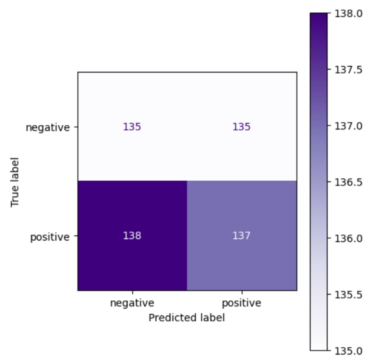


Figure 3: Evaluation Metrics

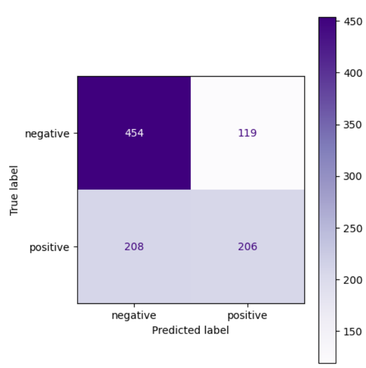
4.1 Positive/Negative aspects

	precision	recall	f1-score	support
negative	0.62	0.42	0.50	374
positive	0.68	0.83	0.75	565
accuracy			0.67	939
macro avg	0.65	0.62	0.62	939
weighted avg	0.66	0.67	0.65	939

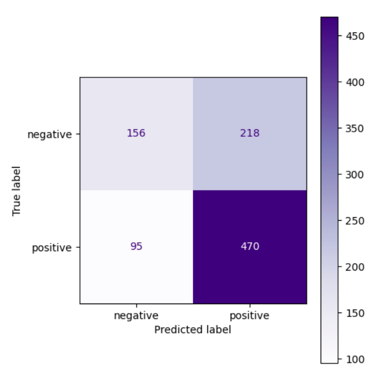
(a) Evaluation Metrics - Positive

	precision	recall	f1-score	support
negative	0.69	0.79	0.74	573
positive	0.63	0.50	0.56	414
accuracy			0.67	987
macro avg	0.66	0.64	0.65	987
weighted avg	0.66	0.67	0.66	987

(b) Evaluation Metrics - Negative



(a) Confusion Matrix - Negative



(b) Confusion Matrix - Positive

Those metrics improved for the classification of the subset of sentences associated with the main positive and negative aspects discovered in section 3. The classification results for **positive sentences** indicate moderate model performance, with an overall accuracy of 67%, meaning the model correctly predicts sentiment 67% of the time. The precision, recall, and F1 scores differ between the two classes: For the positive class, precision is higher at 0.68, and recall is strong at 0.83, meaning that the model captures 83% of the actual positive cases, leading to a more robust F1 score of 0.75. While for the negative class, precision is lower 0.62, meaning that 62% of the predicted negative cases were correct. Recall is lower at 0.42, indicating that the model correctly identifies only 42% of the actual negative cases, resulting in an F1 score of 0.50.

The classification results for **negative sentences** reflect moderate model performance too, with an overall accuracy of 67%, meaning the model correctly predicts sentiment 67% of the time. However, there is a noticeable difference in performance between the two sentiment classes: for the negative class, the model performs well, with a precision of 0.69 and a high recall of 0.79. This means the model correctly identifies 79% of actual negative cases, leading to a solid F1 score of 0.74, indicating a strong balance between precision and recall. For the positive class, performance is weaker, with a precision of 0.63 and a recall of 0.50. This means the model correctly identifies only 50% of the actual positive cases, resulting in a lower F1 score of 0.56. This suggests the model struggles more with accurately identifying positive sentiment.

The metrics are even higher when we apply the 90th percentile instead of the third quartile. The improved metrics can be attributed to the selection of sentences containing the most influential aspects (both positive and negative), resulting in enhanced performance. However, some errors persist because the sentences being classified are associated with the original label of the entire review. This does not necessarily mean that the sentiment of an individual sentence must align with the overall sentiment of the entire review, even though it does in most cases.

4.2 Further Analysis

To investigate the discrepancies between the original label and the predicted one, I have extracted and analyzed some sentences, with a focus on the features that are common for both aspects groups. A relevant discovery was that some reviews were wrongly classified in the dataset as we can see in the example:

1003	Hey, so. I started to use this app so i could communicate with others, especially yters who have discord servers. As a mistake, I misspelled my email by one letter, so i went to change it because I needed to verify it to be able to message in some servers ,and since the email is incorrect, it wouldn't send the message to my gmail/mail for verification. As I went to correct my email and hit save, it would give me a message. The message simply said something along the lines of, "email has already been registered." I'm not sure what it meant by this, because i never verified my actual email just yet. Thus, once i hit save it wouldn't change it. I'm currently stuck with an incorrect email that can't be verified nor changed to the right one, because it's misspelled so it wont send. Is there anything I can do to be able to change my email?? Oh, And i'm on mobile too... if that has something to do with it. What exactly does the message "Email already been registered" mean. My email hasn't been able to be added yet!! And I haven't given anyone access to it either.	positive
1044	I recently bought an Oculus Quest 2 to try and play VR games on PC, so I installed the Oculus app on the desktop & it said that it required a Facebook account. So I'm like okay let me create an account on Facebook so I can start playing. I created one on my phone then tried to log in to Facebook on PC but I forgot the password like 2 times & now it's making me verify my account. I'm like okay...then it tells me to verify using E-mail. I clicked the E-mail that was associated with that account and EVERY TIME I try to get a verification for my E-Mail it says there's an error, never sends the verification and says to retry later. 2 days later I'm still trying to log in and the same thing shows up. And from what I've saw there's no way to contact Facebook about this issue. So basically I bought the Oculus Quest 2 for no reason cause I can't even log in when that's the only possible way of being able to connect it to your PC. This is kind of sad when all I'm trying to do is enjoy my Oculus for PC gaming but I can't cause Facebook decides to not work for me. Please fix this soon, it's making me more impatient as each day passes. ☹	positive
1142	Just need to add a Mac app and this would be the perfect application. If you are wondering this is a cryptographically solid application with string encryption between devices. Once you achieve three dot verification the chances of a man in the middle attack are all but eliminated.	positive
1175	I have What's App on an iPhone and the layout of the chat screen positions the call icon in the upper right corner. The phone frequently places inadvertent calls when I put it in my pocket, or accidentally lightly touch the icon when picking up the phone. Even closing the app and locking the screen doesn't prevent calls if a notification appears on the lock screen, which seems to wake up the app. The developer needs to move the call icon off the chat screen, or add an option of enabling a secondary verification for calls.	positive
1405	I don't know who thought it would be a good idea, but the Google verification slide needs to get out and GO. It has never worked properly on mobile devices and they don't seem to be in any hurry to fix it. The security on this app was perfectly fine. I don't need to sit here for TEN MINUTES(!!!) and tap pictures only for the app to say I'm not doing it right. One star off for that. The fact that Do Not Disturb always seems to be broken does not help. It *does* work, intermittently, but it feels like every time it gets fixed a new update drops and it breaks again. One star off for that as well. 3/5 overall. This app is perfect otherwise.	positive
1518	Dear discord,Right now I'm extremely mad and furious, you want to know why you may ask? Well I figure there is a lot of bugs in your app that you may not know of. First I made a discord account that I loved with a group and a lot of a friends, I also was attending groups and having fun until I was wanting to invited people to my group. It showed up a picture saying this "Somethings going on here" it also says this "we've detected something out of the ordinary going on. To continue using discord, we will need you to verify you account" so I was like no big deal right? So I did everything. I put in my number, I put in the verification number, until it got down to my password. I was like "oh I know my password in a heart beat. So I put the password in. It was correct to I thought when I hit done it will bring me back to the page with my friend. When I hit done, it brought me to are you a robot thing. So I did the task and I was done so I hit verify. And guess what... I hit done again.. it loaded.. and did nothing. I was having a full panic attack. So I deleted the app and did the whole thing over again. Yes all over again. Same thing happened. So I don't know if it's a bug or you guys want people to get so angry. So if u get things please update or do something so you don't get other people mad to. Thank you..?-sincerely Audrina	positive

Figure 6: Wrong Label

It's clear that, even if the sentiment is "positive", the reviews are actually negatives. This could have brought Born to learn "verification" as a positive relevant word, even if appears mostly in negative reviews (that have been labeled as positives).

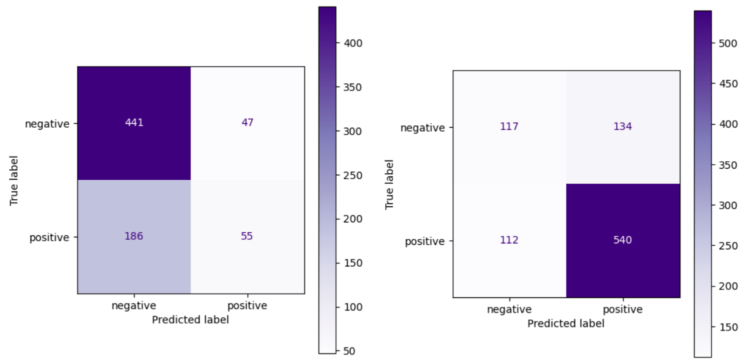
At the end we can say that, even if those words are relevant for both aspects with different weights in positive or negative, we can assign the words to the group aspect for which the weight is higher, excluding the feature in the other group aspect. This is true for most of the words except for some of them like 'verification', 'save' for which the sentiment is mostly negative, and for 'messaging' and 'group' for which the classification has given more positives than negatives because of the wrong associated labels with the reviews. Anyway, assigning these features only to positive or negative aspects and changing the respectively sentiment, has improved the classification result, as we can see in the plots:

	precision	recall	f1-score	support
negative	0.70	0.90	0.79	488
positive	0.54	0.23	0.32	241
accuracy			0.68	729
macro avg	0.62	0.57	0.56	729
weighted avg	0.65	0.68	0.64	729

(a) Evaluation Metrics - Negative

	precision	recall	f1-score	support
negative	0.51	0.47	0.49	251
positive	0.80	0.83	0.81	652
accuracy			0.73	903
macro avg	0.66	0.65	0.65	903
weighted avg	0.72	0.73	0.72	903

(b) Evaluation Metrics - Positive



(a) Confusion Matrix - Negative (b) Confusion Matrix - Positive

5 Related Issues

The main issues arise in the detection of candidate aspects. The methodology adopted works fairly well but has some limitations. Since POS tagging is performed on the entire set of reviews, if a candidate feature appears more than once in the review with different grammar dependencies, I was unable to match it correctly in the sentence where it functions as a noun, either as a subject, object, or compound noun. Additionally, some words that are not nouns are detected due to misspellings in the review text or the use of abbreviations in informal text. For example, ”atleast” is a misspelling of ”at least,” and ”asap” is an abbreviation of ”as soon as possible.” Another example is ”barley - NOUN,” which is a real word, but in this specific context, it’s a misspelling of ”barely - ADV.” Due to these typing errors, SpaCy’s POS tagging predicts the wrong part of speech, incorrectly classifying some words as nouns when they are not and leading to not relevant candidate aspects. Another issue regarded the negation in the sentence and how to deal with them.

6 Conclusion

In conclusion, the results obtained are satisfactory, and a correlation between the sentiment of the aspects and the overall sentiment of the entire review has been identified. Moreover, Born’s algorithm performs well for classification purposes. However, there are areas for improvement, particularly in candidate aspect detection, where misspellings have led to errors in the POS tagging analysis. Using different techniques for detecting candidate aspects, such as Latent Dirichlet Allocation (LDA) or Word Embedding, could enhance the results. Further analysis can also be conducted on the aspects that cause a mismatch between the original label and the predicted label to determine whether other reviews were misclassified in the original dataset, allowing for sentiment adjustments accordingly.

7 References

Emanuele Guidotti and Alfio Ferrara. Text Classification with Born's Rule. Advances in Neural Information Processing Systems, 2022

Schouten, K., & Frasincar, F. (2015). Survey on aspect-level sentiment analysis. IEEE Transactions on Knowledge and Data Engineering, 28(3), 813-830