**Business Insights**

1. What are the most frequent words associated with negative reviews? What are the main areas of underlying problems that are reflected in these words?

Counting was conducted to find the most frequent words in negative comments. Based on the code’s output, words of “not” have occurred most of 2067 times, and “but” had the second highest frequency. Generally, there have been higher proportions of negative sentiments extracted than positive comments.

There were three main areas of underlying problem reflected by the words. Firstly, the comments revealed a considerable level of food- and product-related disappointment, such as “not buy”, “not good”, “not taste good”, indicating negative sentiment. Although there were words like “tasty” and “good”, they mostly appear in phrases of “not good” or “not tasty”, which all pointed to dissatisfaction. Secondly, the comments could infer about product-quality-related issues as there were words of “way too weak” and “stuck”, which could be malfunctions. Also, the word “tiny” could be commenting on the product size. Lastly, the comments could reflect on the difference between customer’s expectation when ordering versus what they actually received. For example, words like “not worth money” and “would not recommend” could be suggesting that products were related with misleading reviews or customers’ complaints about their order that did not measure up their expectations.

A screenshot of a computer screen

Description automatically generated

1. Are there recurring phrases that indicate product defects or poor customer service?

The code has preprocessed text by expanding contractions from words like “isn’t” to “is not”, and the table has shown that the most frequently phrases were associated with product defects and poor customer services using get\_ngrams function. For example, there were negative bigrams of “not good” and “not buy” that indicated bad product quality and; trigrams of “not worth money”, “would not eat”, “would not recommend” showed that customers were dissatisfied of their user experiences.

1. Based on the recurring negative-review patterns, what specific improvements or strategic changes would you recommend to address these concerns?

Strategic improvements should be implemented correspondingly to the three main underlying areas, which were food-related disappointment, product quality issues, and user experiences (that the products received did not match with customer’s expectations).

The comments have shown high-frequency words of “flavour”, stale”, “tastes bad” that suggested food taste issues. Therefore, in the food-preparing process, a stricter quality control should be implemented to ensure ingredients’ freshness. Besides, recipes could be adjusted to reduce bitterness or develop more ingredients that could produce better flavors.

When targeting product quality issues, there were comments like “way too weak” or “not working”. Thus, a better product testing should be implemented before the product was launched or sent out to customers, or to use high-quality materials to better sustain the lifetime of the machine. Furthermore, longer warranties or more considerate return policies should be made so that customers could feel “backed up” when using their product, which could also enhance customer satisfaction to after-sale services.

Lastly, to better manage consumer expectations and user experiences, companies should provide a better match between product’s picture and reality. For example, companies should be more precise in their product descriptions and comprehensive product images that could give customers’ deeper understanding to the product before purchases. Or companies could respond more quickly in customer service with clear return and refund policies to increase efficiency.

**Challenges:**

One challenge that we have encountered was that as shown in the output, single-word tokens have achieved better performance on validation data than bigrams, which was the opposite to our initial expectations. This could be caused by several reasons. Firstly, bigrams did not necessarily have stronger sentiment than single-word tokens. For example, “good” is equivalent to “taste good” in the sense that they were all positive comments. Conversely, some single-word could have stronger sentiment and can be better captured than bigrams. Besides, when using single tokens, each word will occur many more times in training data than bigrams were individual bigrams appeared less frequently. This could reduce generalization of the data. Furthermore, bigrams could perform better in larger dataset, but could be the opposite with limited input because with the small set of training reviews, the model memorized the pattern instead of learning the pattern, which contributed again to overfitting and weak generalization.

Another challenge was overfitting risk. Since the model was learning patterns from training data – a limited dataset, this could make the model lass general and inaccurate in predicting larger real-world data. Besides, because of the small dataset with limited input, the model could assign high weights to irrelevant words, resulting in large variance. It was also challenging in searching for the best hyperparameter in the small dataset. Nevertheless, early stopping was implemented to prevent the issue of overfitting since it could restore the best epoch where the validation loss was the lowest and stop training before overfitting.

Lastly, with analyzing word frequencies was that when comparing between simply counting the words or assigning model weights, either method would have visible limitations. With counting, it’s noticed that frequently words didn’t always mean strong sentiment as some could appear often but have neutral meanings, like “product” and “buy”, which did not draw conclusions to any of the business insights. Besides, some comments could have sarcasm: words like “good” could be positive as a single token, but it could become “not good” if analyzed in bigrams; “smells like cat food” was actually a sarcastic and negative comment, but such expressions could be hard to classified. With word weighting, neural network weights were usually too complicated to be manually interpretable, which was challenging. And since there are randomness in every run, there could be different output every time.