Assessing the importance of user-defined product dimensions in reviews (Github link)

Louis-Charles Généreux

Abstract

In this paper, I present a novel NLP-based approach to identifying the importance of different product dimensions in a user's experience. This approach first identifies feature words that characterize experiences as positive or negative across different product categories, then assesses the extent to which features relate to each dimension. This approach results in interpretable product category comparaisons, highlighting which dimensions most matter to customers. The results from this analysis can inform marketing decisions by manufacturers or retailers, and could be reproduced across domains where review text is available (e.g., tourism, customer service, retail).

1 Introduction

With an ever-growing quantity of reviews available online, companies can now use new NLP-based approaches to better market their products and services. Rather than rely on expensive (often biased) surveys or focus groups to improve their marketing strategies, they can mine user-generated reviews to identify common themes across strong or poor reviews. Alternatively, they can also generate automated responses to comments (taking context and topic in consideration).

In this paper, I develop an approach in which I test the relative importance of different product dimensions (e.g., aesthetic, price, ease of use, performance) across various categories (e.g., toys, food, appliances). This allows me to make claims like: 'shoppers of appliances are price sensitive and do not care about aesthetics' while 'toy and game shoppers value sensory appeal and ease of use'. Such insights may then inform the communications and marketing strategies for manufacturers and retailers of these products.

To compare product categories, I first gathered reviews across 6 different categories. After preprocessing reviews of each category, I built 6 separate classification models whose goal was to identify positive and negative reviews. I extracted the top 100 word features for each model (along with their relative importance). I then iterated through each predictor, assessing its similarity to a set of dimension-specific nearest neighbor words. After aggregating results across all predictor words (and weighing by their importance), this resulted in a dimension importance score for each product * dimension combination.

This approach differs from current topic modeling and text mining methods in that it is not fully unsupervised. Instead, it gets inspiration from some topic matching techniques, where users can define topics ahead of modeling. Using defined topics to create product summaries improves interpretability of text-mining outputs and may help generate buy-in from industry executives (who come with their preconceived notions).

By generating clear and concise product summaries across user-defined dimensions (from hundreds of thousands of reviews), we equip marketing executives with a new tool to track what their customers value (and to prove/ disprove product-related hypotheses). I look forward to following new iterations of this approach: use across a wider set of industries, including the time component (seeing how importance across dimensions has varied over time for a given product), etc.

2 Related work

Throughout the development of this tool, I read about: (A) topic modeling, (B) model interpretability and (C) topic matching. Below are comments on their relation to my work:

(A) Topic modeling. Prior to developing my product evaluation approach, I attempted to identify topics across a corpus of product reviews. I used Linear Discriminant Analysis (LDA) to assign each review to one topic, then observed the frequency of different topics across product categories. While the approach generated results very quickly, I noted a few major drawbacks. The first one is that upon interpretation of topics, I notice major overlap across topics (many repetitions in the top 20 most relevant words to each topic). Second, I find certain dimensions to be difficult to interpret (the same topic includes themes which I would separate such as taste and time). Finally, some of the resulting topics do not even refer to some dimensions which I would like to account for.

Given that I aimed to understand the importance of different user-defined dimensions, a purely unsupervised approach to topic modeling was not desirable.

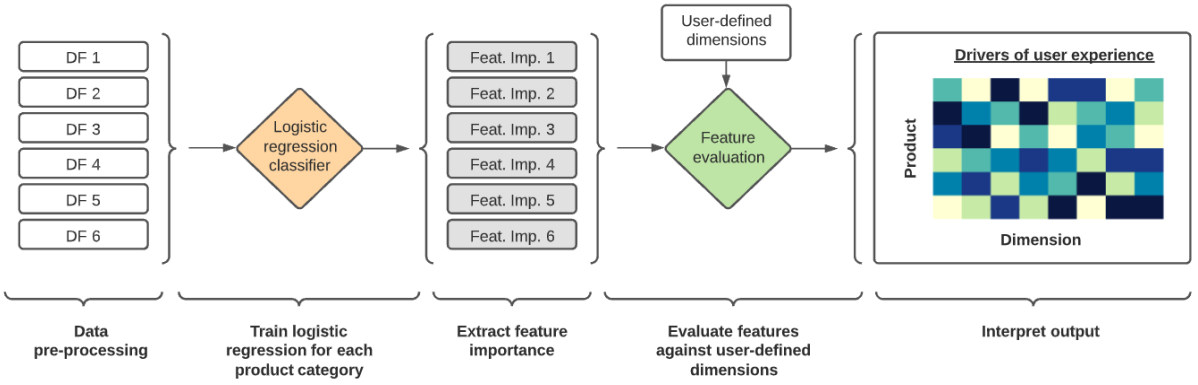


Figure 2. Methodological details

(B) Model interpretability. Another ambition of mine when developing this framework was to identify themes that make or break product reviews across categories (as opposed to just interpreting word frequency across categories). By building classification models for each category (to classify each review as positive or negative), I aim to identify the words that matter most in understanding users' experiences across categories.

While Local Model-Agnostic Methods like LIME allow users to identify words in one text entry which most contributes to a classification, I decided to not use this method at scale (I won't aggregate LIME word importance results across all observations). First, because it would require that I iterate through every text entry (non-scalable), second because these methods' local sampling may generate unstable results.

Instead, I aim to make my approach interpretable at the macro level (product level) as opposed to observation by observation. I hope that visualizations and summary of my work can be interpretable by non technical individuals.

(C) Topic matching in marketing. Topic matching is increasingly used in the field of travel and hospitality. For example, Zhang, Qiao, Yang and Zhang (2020) have used topic matching to prove that hotels responding directly to customer complaints (addressing the same topic as was mentioned in the complaint) receive better reviews. Their methodology made use of user-defined themes as opposed to ones generated through unsupervised learning. Reading about this approach, which contrasts from unsupervised approaches mentioned in (A), inspired me to define a set of hypotheses to be tested in my framework.

3 Dataset

Implementation of my proposed approach used 6 separate datasets, all corresponding to reviews of a different Amazon product category. Each dataset consists of both text entries and a star review (ranging from 1 to 5). Data preparation steps prior to modeling included (a) the conversion of numeric reviews to binary classes (1 and 2 stars as 'low', 4 and 5 stars as 'high') and (b) text pre-processing (data cleaning, removal of stop-words, etc).

Figure 1. Dataset statistics

	Appliances	Fashion	Gourmet food	Magazine subscriptions	Musical instruments	Toys and Games
Observations	160,720	343,594	211,900	32,502	369,776	244,002
Words	8,073,989	10,050,838	8,882,375	1,671,134	18,531,520	12,524,412

4 Method

As is highlighted in **Figure 2**, building this framework can be achieved in 4 distinct steps.

(1) Data pre-processing where the dataset for each product category is prepared for logistic regression (as described in section 3).

(2) Training of logistic regressions, classifying each review as positive or negative. For this exercise, we use unigrams only as features (so that their distance to other words can be evaluated in subsequent steps.

(3) Extracting feature importance from each logistic regression classifier (top 50 positive predictors and top 50 negative predictors). Results from each classifier are appended to a csv file containing the word and its weight in the classification model.

(4) Evaluate features against user defined dimensions. For this exercise, I have defined 8 different dimensions (aesthetics, smell / touch / taste, price, fit / size, delivery, entertainment, ease of use and performance). For each of these dimensions, I have defined 3 synonyms', for example ['functional', 'defective', 'operational'] for the performance dimension. Using GLOVE pre-trained word embeddings, I identify the 5 nearest neighbors to each synonym, resulting in at most 15 'reference words' which best capture the essence of the dimension.

Once dimensions have been defined, I iterate through each product category, assessing the relative importance of each product dimension in its reviews. This is achieved through the following algorithm, which iteratively calculates the similarity between a

product category's word features (generated through logistic regression) and reference words for each dimension, then weighs the similarity based on the feature's importance in the model.

Algorithm

```
product_category = []
dimension_name = []
sim_score = []

# For all product categories
for product in product_category:
  # list of features from logit (unique to product)
  for feature in list_features[product]:
    # Same dimensions tested for each iteration
    for dim in dimension_list:
      weighted_simil = []
      # Reference words for dimension
      for reference in reference_words[dim]:
        # Calculate similarity as inverse of dist
        simil = 1 / dist(feature, reference)
        weight_sim = feature_weight * simil
        weighted_simil.append(weight_sim)

    # Store after iterating through dimension
    product_category.append(product)
    dimension_name.append(dim)
    sim_score.append(sum(weighted_simil))
```

Once aggregated at the product and dimension level, we have the weighted similarity between each product and the user defined dimensions (weighted based on the feature importance of words in logistic regression). For comparison purposes and interpretability, these data can be scaled or given a value based on their rank (e.g., #1, #2, … , #7, #8 dimensions for category).

5 Results

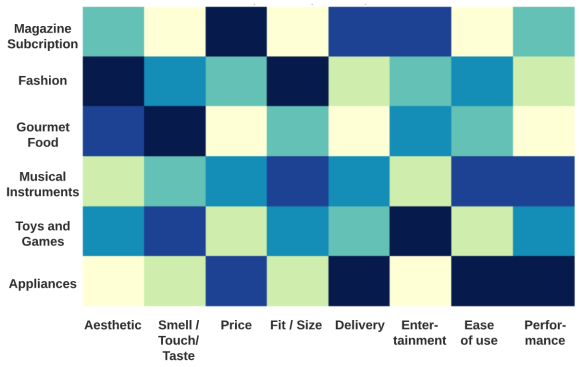
Figure 3. Logistic regression results

	Appliances	Fashion	Gourmet food	Magazine subscriptions	Musical instruments	Toys and Games
Accuracy	0.91084	0.91283	0.90824	0.88448	0.90995	0.92314
F1	[0.911, 0.91]	[0.913, 0.912]	[0.908, 0.908]	[0.887, 0.882]	[0.91, 0.91]	[0.923, 0.923]
Precision	[0.906, 0.916]	[0.91, 0.916]	[0.907, 0.909]	0.873, 0.897]	[0.911, 0.909]	[0.923, 0.924]
Recall	[0.917,0.905]	[0.917, 0.909]	[0.91, 0.907]	[0.901, 0.868]	[0.908, 0.912]	[0.924, 0.922]

After having aggregated results as highlighted in section 4 and after having scaled product * dimension scores, we can begin to interpret results through the heatmap below.

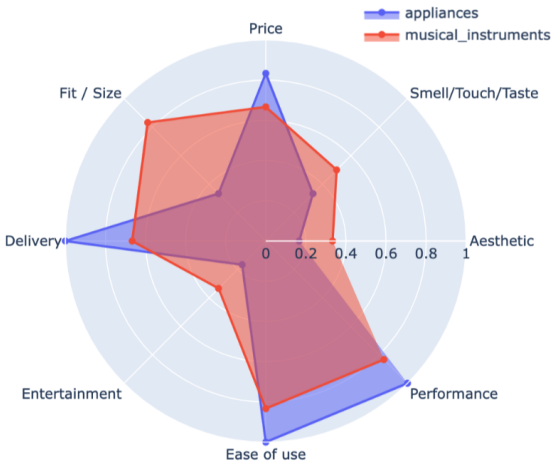
One can look at the intersection of a product and a dimension to assess its relative importance in determining if the review is positive or negative (a darker color is indicative of higher weight).

Figure 4. Scaled summary results



A marketer in charge of launching a successful magazine campaign could be interested in knowing that price is the most important dimension in defining user experience for this category, ahead of a punctual delivery and the magazine's entertainment power).

Figure 5. Scaled results (product comparison)



Also, a comparison between 2 products is possible, for example appliances and musical instruments. While users of these 2 products highly value 'ease of use' and 'performance' (non defective), we notice that punctual delivery and price are more important for appliances purchasers, while fit / sizing and overall feel is more important for musical instrument buyers. Such product comparisons could be very useful for online retailers which carry different product categories and generate targeted word ads online.

6 Discussion

This novel approach to identifying the importance of different product dimensions in a user's experience can have multiple industry applications:

- Side by side product comparison to generate marketing campaigns. This application can even be extended to comparaison within product categories (e.g., fiction vs nonfiction for books, city hotels vs resort-type hotels for hotel chains).
- Time series evolution of consumer tastes by comparing dimension scores for a product before/ after a pre-defined date. Such a use-case could be helpful for marketers attempting to generate a new marketing message for an old product (e.g., consumer goods ads during the holiday season, during a Superbowl spot).

While this project's objective was reached (generating actionable insights from a vast corpus of reviews), some work remains to be done on this topic. First, it should be noted that since the framework currently evaluates products on user-defined dimensions (to test some of my own hypotheses as to what drives satisfaction), we may be missing some un-defined dimension. A remedy to this would be to identify the most frequently used words across reviews (excluding stopwords), then cluster the words based on their embeddings, potentially revealing some new dimensions. Second, several parameters could be tweaked in the framework (e.g., the number of features that are stored after logistic regression, the number of synonyms used to define a dimension), further experimentation would be required to define these changes' effects on interpretability. Finally, some A/B testing in online ad messaging could help confirm or disprove some of the production * dimension importance conclusions that were generated from the framework.

References

Github repo: link

Rania Albalawi, Tet H. Yeap and Morad Benyoucef. 2020. *Using Topic Modeling Methods for Short-Text Data: A Comparative Analysis*. https://doi.org/10.3389/frai.2020.00042

Marco T. Ribeiro, Sameer Singh and Carlos Guestrin. 2016. *"Why Should I Trust You?" Explaining the Predictions of Any Classifier.* https://arxiv.org/pdf/1602.04938.pdf

Molnar, Christoph. 2019. *Interpretable machine learning. A Guide for Making Black Box Models Explainable,* Chapter 9: Local Model-Agnostic Methods. https://christophm.github.io/interpretable-ml-book/

Ian Tenney, James Wexler et al. 2020. *The Language Interpretability Tool: Extensible, Interactive Visualizations and Analysis for NLP Models.* https://arxiv.org/pdf/1602.04938.pdf

Xiaowei Zhang, Shuchen Qiao, Yang Yang and Ziqiong Zhang. 2020. *Exploring the Impact of Personalized Management Responses on Tourists' Satisfaction: A Topic Matching Perspective*. https://doi.org/10.1016/j.tourman.2019.103953

Jianmo Ni. 2019. *Amazon Reviews.* *https://nijianmo.github.io/amazon/index.html*

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. *GloVe: Global Vectors for Word Representation.* https://nlp.stanford.edu/pubs/glove.pdf