Frogs into princes: A generative model to understand the success of product descriptions

Takehiro Takayanagi[♣], Bruno Charron[♢], Marco Visentini-Scarzanella[♢]
[♣] The University of Tokyo, School of Engineering, Tokyo, Japan
[♢] Amazon, Tokyo, Japan
takayanagi-takehiro590@g.ecc.u-tokyo.ac.jp
{bcharro, marcovs}@amazon.co.jp

Abstract

In the dynamic marketplace, vendors continuously seek innovative ideas for new products and ways to improve existing ones. These ideas can be uncovered by analyzing text data, such as product descriptions and customer reviews. However, the ever-increasing volume of text data poses a challenge in extracting meaningful insights. Therefore, this study addresses the challenge of extracting actionable insights from the growing volume of text data, with a specific focus on product descriptions. To this end, we investigate two primary research questions: the predictive power of product descriptions for product success, and the capability of style transfer to highlight the successful factors of these descriptions. In response to the first question, our findings validate that product descriptions are indeed reliable indicators of product success. Addressing our second question, we propose a Successful Style Transfer Variational Autoencoder (SST-VAE), a VAE-based language model designed for effective successful style transfer. Qualitative analysis indicates that the SST-VAE effectively enables successful style transfer conditional on a given label. In addition, case studies suggest that the proposed approach could be useful in gaining insights about product success, by highlighting key factors that may contribute to their success. On the other hand, our approach confronts issues such as hallucinations and the need for factual accuracy. These challenges underscore the necessity for continued research in the field of e-commerce natural language processing.

Keywords: generative model, product description, natural language processing

1. Introduction

In the dynamic marketplace, vendors continuously seek innovative ideas for new products and ways to improve existing ones. This valuable knowledge could be revealed through the analysis of textual data, including product descriptions and consumer reviews. Yet, the rapid expansion of textual data volume poses a significant challenge in manually analyzing them. Accordingly, natural language processing (NLP) is utilized to extract meaningful insights from textual data, which has resulted in extensive research in the field of e-commerce NLP.

In the domain of e-commerce NLP, a significant body of work focuses on product descriptions, recognizing their crucial role in providing a competitive customer experience (Wang et al., 2017; Chen et al., 2019; Chan et al., 2019; Zhang et al., 2019; Zheng et al., 2018). In this realm, a considerable portion of the current research focuses on text generation for product descriptions. This focus has given rise to unique challenges, including personalized generations (Chen et al., 2019) and fidelity-oriented generations (Chan et al., 2019). However, there has been little research aimed at extracting valuable insights about product success.

Addressing this research gap, the primary aim of this study is to extract insights regarding product success from product descriptions. Following this direction, two key questions emerged.

Initially, there is uncertainty regarding the extent to which product descriptions contain insights into a product's success. Figure 1 presents a comparison between the description of a successful product (which has a higher rating) and that of an unsuccessful product (which has a lower rating). From this figure, it remains unclear to what extent these descriptions capture information pertinent to product success, especially given that product success is often influenced by various factors, including external trends and events. Therefore, it is worthwhile to determine the degree to which these descriptions encompass relevant information. This consideration guides us to our first research question: "To what extent can product descriptions accurately indicate a product's likelihood of success?"

The second question emerges from the difficulty of manual dataset collection. Typically, extracting product success factors from product descriptions entails annotating these successful elements in product descriptions. Nevertheless, as Figure 1 illustrates, identifying the key elements that contribute to a product's success from its descriptions is quite challenging, which significantly complicates the manual development of a dataset. In response to these complexities, we consider that a style transfer approach could mitigate these challenges (Jin et al., 2022). It enables comparisons between original texts and those transformed into a successful style without the challenges of manual annotation.

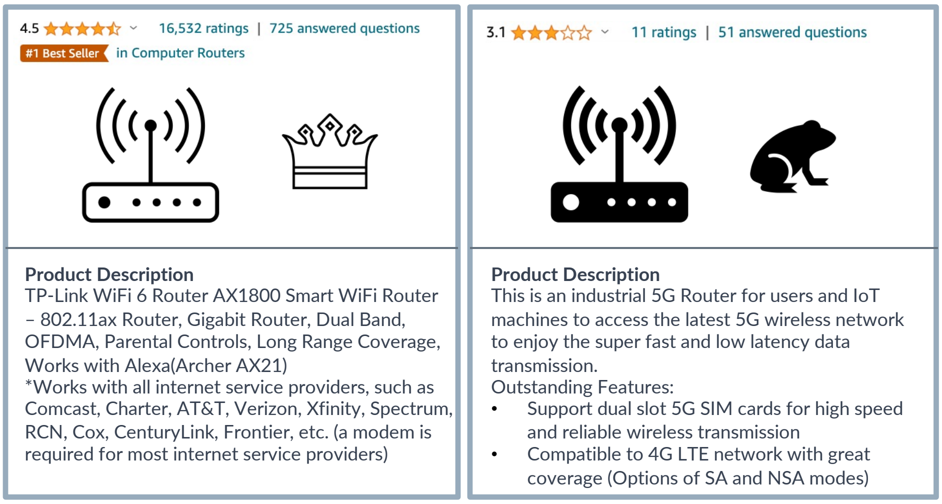

Figure 1: Two Product Examples with Similar Features but Varying Success: The Product On The Left Has A Higher Customer Rating, While The Product On The Right Has A Lower Customer Rating.

These comparisons will be beneficial in uncovering product success factors. This leads to our second research question: “Can we transform text descriptions from unsuccessful to successful ones, or vice versa?”

In summary, we formulated the following research questions

- RQ1: To what extent can product descriptions accurately indicate a product’s likelihood of success?
- RQ2: Can we transform text descriptions from unsuccessful to successful ones, or vice versa?

This paper addresses these research questions. Firstly, we examine the potential of product descriptions as predictors of product success in response to the first research question. To this end, we introduce a success prediction task from product descriptions to assess the extent to which product descriptions can predict success.

Subsequently, we introduce a novel task involving the style transfer of product descriptions for the second research question. To address this challenge, we propose a Successful Style Transfer Variational Autoencoder (SST-VAE), a model designed to generate product descriptions conditional on the product’s success. We assess the proposed model’s output in qualitative analysis and case studies based on two perspectives: first, its ability to highlight factors of product success; second, its effectiveness in enhancing original descriptions through successful text transformations. Our results indicate that the proposed model successfully transforms the product descriptions from unsuccessful to successful and vice versa. In addition, our results suggest that the successful style transfer method can be effective in extracting insights about products by emphasizing the potential factors that could contribute to their success. On the other hand, we also reveal several issues, such as hallucinations, particularly when applied to enhance the product descriptions themselves, laying out a path for future research opportunities in this domain.

2. Related Work

2.1. Text Style Transfer

Text style transfer, an important task in natural language generation, focuses on modifying attributes like sentiment and politeness in text. It has regained notable attention in natural language processing due to the impressive results achieved with recent deep neural models (Jin et al., 2022). The challenge in text style transfer lies in the ambiguity of “style” in NLP and many works are tackling the problem. Some works deal with the sentiment of a text as a style of the text. Hu et al. (2017) proposes to control the sentiment of a text by using discriminators to reconstruct sentiment and content. Also, John et al. (2019) tackles the problem of disentangling the latent representations of style and content by proposing a multi-task loss to achieve the style transfer in terms of sentiment on non-parallel corpora. Additionally, Vasilakes et al. (2022) advances the field by enabling style transfer specifically in the realms of negation and uncertainty. Despite the extensive research in NLP style transfer, no existing study has explored text style transfer in the context of a product’s success.

2.2. NLP in E-commerce

The NLP research in the e-commerce domain has shown considerable interest in product descriptions, with numerous studies addressing challenges such as text generation and attribute extraction (Wang et al., 2017; Chen et al., 2019; Chan et al., 2019; Zhang et al., 2019; Zheng et al., 2018). Especially, text generation task draws attention as quality product descriptions are important for providing a competitive customer experience in e-commerce stores. For instance, Wang et al. (2017) introduces a statistical model capable of producing accurate and fluent descriptions of product attributes. Further advancing the field, Chen et al. (2019) delves into generating personalized product descriptions using a knowledge base and neural networks. Additionally, Zhang et al. (2019) explores a pointer-generator neural network to generate product descriptions where the output patterns are controlled. Chan et al. (2019) emphasizes the fidelity of content in their approach to product description generation. Despite these advancements, most text generation research in this area has primarily focused on automating the product description writing process. There has been little emphasis on obtaining insights from descriptions or enhancing them.

Recognizing the importance of extracting meaningful insights from these text data, we propose a novel approach, employing a style transfer approach to gain insights into what drives product success.

3. RQ1: Can we predict products' success from product description?

Figure 2: An Illustration of the Success Prediction Task.

3.1. Success Prediction Task

The initial research question we address is, "Can we predict a product's success from its descriptions?" To tackle the question, we introduce a success prediction task based on product descriptions, as depicted in Figure 2. In this task, we train a natural language processing model to predict whether a product will be successful or not given product descriptions. The reasoning behind establishing this task for RQ1 is that if the model can predict success to a reasonable degree, it indicates the presence of significant information within product descriptions that correlates with the product's success. In our experiments, we utilize a pre-trained BERT (Devlin et al., 2019) model to predict product success.

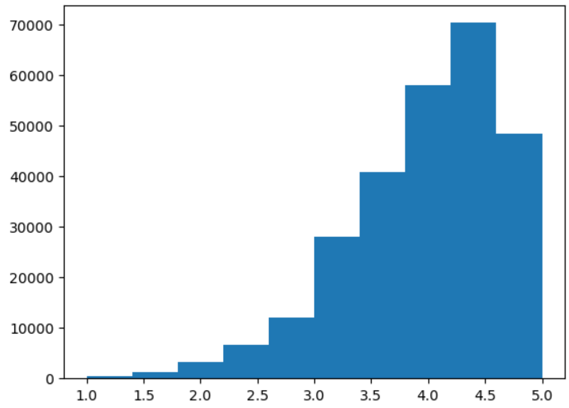

Figure 3: Histogram Displaying the Average Ratings Following Data Preprocessing.

3.2. Evaluation Settings

In our experiments, we utilize the Amazon Public Dataset [1], which comprises a substantial collection of product data from Amazon. We focus our experiments on a single category, as we consider that the factors driving success vary significantly across different categories. Specifically, our experiments employ data from the Electronics category. For data processing, we exclude products with less than 5 ratings as well as those with duplicated descriptions, amounting to a total of 269,469 products. We define a product's success based on its average rating. Specifically, a product is labeled as successful if its average rating exceeds a certain threshold (for our study, thresholds are set at 3 or 4 out of 5 stars), and as unsuccessful if the average rating falls below 3. Figure 3 shows the distribution of average product review ratings.

We implement two data-splitting strategies to mitigate potential data leakage. The first approach is a random split, where data is divided randomly into training and test sets. In the random splitting strategy, we observe some instances where product descriptions share identical beginnings as shown in Table 1, and that the model often assigns the same labels to them. This raises concerns about data leakage, where the model's predictions for the test dataset might be influenced by the recurring text in the training dataset. To address this, we also adopt the group stratified data splitting strategy. Here, products sharing the first x characters (a predetermined threshold) in their descriptions

[1] https://cseweb.ucsd.edu/~jmcauley/datasets.html#amazon_reviews

Table 1: Examples of Two Product Descriptions with Identical Initial Sentences (Italicized) to Highlight Shared Content.

	Description
D1	*Bring your digital camera back to life with a new battery. Make sure you never miss another once-in-a-lifetime moment by having a new*, battery specifically designed for your Canon Rebel T2i T3i T4i T5i digital SLR camera. BM Premium rechargeable batteries are engineered to meet or exceed OEM specifications and feature the latest battery technology, including advanced circuitry, voltage regulation, and thermal circuit protection. BM Premium batteries include a one-year warranty.
D2	*Bring your digital camera back to life with a new battery. Make sure you never miss another once-in-a-lifetime moment by having a new*, LPE-10 battery specifically designed for your Canon Rebel digital SLR camera. LPE10 rechargeable batteries are engineered to meet or exceed OEM specifications and feature the latest battery technology, including advanced circuitry, voltage regulation, and thermal circuit protection. This battery includes a one-year warranty.

Table 2: Results of the Success Prediction Task.

Splitting strategy	Random	Group shuffle with threshold				
Success label		Higher than 3				Higher than 4
Threshold	-	50	100	300	500	100
Accuracy	0.71	0.69	0.69	0.70	0.70	0.88
F1	0.75	0.74	0.74	0.74	0.74	0.94
Precision	0.72	0.69	0.71	0.72	0.72	0.90
Recall	0.79	0.79	0.77	0.76	0.77	0.97

are grouped together. We then ensure that descriptions from the same group are exclusively assigned to either the training or the test dataset, but not both.

As for evaluation metrics, we utilize accuracy, recall, precision, and F1 score. In the training, we finetune the BERT model for 3 epochs with learning rate 0.001.

3.3. Result and Discussion

The results of the success prediction task, presented in Table 2, indicate that product descriptions effectively predict a product's success, achieving around 70% accuracy in most scenarios. Moreover, this consistency in prediction accuracy persists across various experimental settings and different data-splitting strategies, including variations in the threshold value x.

Additionally, the model achieves greater performance when defining success at a level higher than 4, compared to a threshold higher than 3. This outcome indicates that there are meaningful distinctions within product descriptions that contribute to the successful differentiation between successful and unsuccessful products.

On the other hand, the precise reasons for the model's effective prediction of product success are not clear. We hypothesize two possible scenarios. Firstly, there may be a distinct style of writing in product descriptions that contributes to their success. Secondly, it is possible that successful products possess appealing features, and the model captures these features within the product descriptions. Investigating these hypotheses will be reserved for future research.

4. RQ2: Can we transfer the text description from unsuccessful to successful one or vice versa?

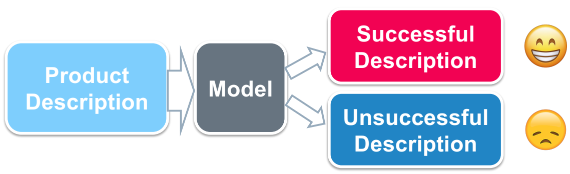

Figure 5: An illustration of the Successful Style Transfer Task.

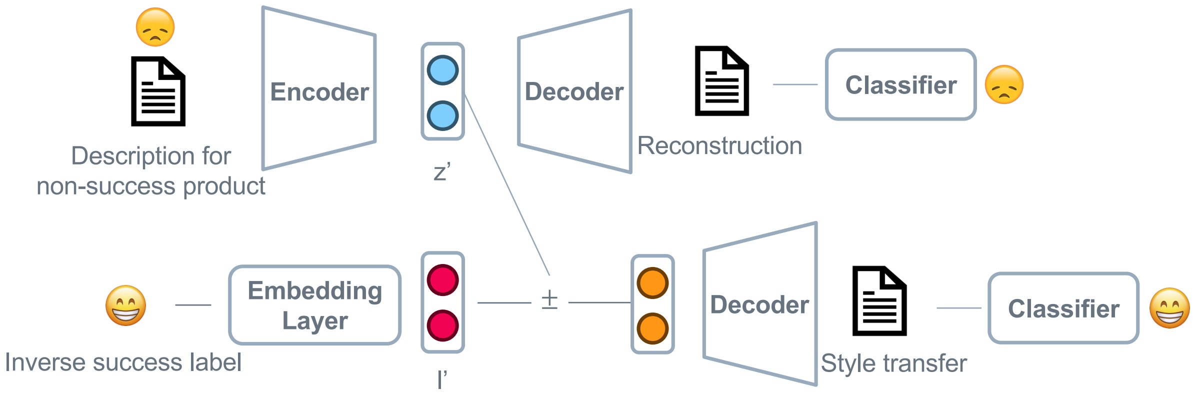

Figure 6: The Overview of Proposed SST-VAE Model.

4.1. Successful Style Transfer Task

Having confirmed that product descriptions are indeed reliable predictors of product success, we then move on to address the second research question: "Can we transform text descriptions from unsuccessful to successful ones, or vice versa?" To validate this question, we introduce a successful style transfer task, where an NLP model is trained to generate either more successful or less successful product descriptions as depicted in Figure 5. The underlying motivation for this task is that by shifting from successful to unsuccessful descriptions or the reverse, we can gain insights into the factors driving product success.

4.2. Method

For the successful style transfer task, we employ VAEs as text autoencoders due to their provision of smooth latent space, facilitating the transfer of texts conditioned on a label (Kingma and Welling, 2014; John et al., 2019). The basic VAE objective (i.e., the reconstruction loss with KL divergence) can be written as follows.

$$\mathcal{L}_{\beta} = \mathcal{L}_E + \beta \mathcal{L}_R \quad (1)$$

$$\mathcal{L}_E = - \mathop{\mathbb{E}}_{q_\phi(z|x)} [\log p_\theta(x|z)] \quad (2)$$

$$\mathcal{L}_R = KL(q_\phi(z|x) \| p(z)) \quad (3)$$

where $\mathcal{L}_E$ represents the reconstruction loss, while $\mathcal{L}_R$ denotes the KL divergence loss. The loss function $\mathcal{L}_\beta$ is a weighted combination of the reconstruction loss and KL divergence, with the β coefficient determining the influence of KL divergence on the overall loss. Typically, $q_\phi(z|x)$ is modeled as a Gaussian distribution with model parameters ϕ, and the re-parametrization trick is employed during training (Kingma and Welling, 2014). Additionally, $\log p_\theta(x|z)$ is a conditional probability of the data x given latent variables z with model parameters θ. For the prior distributions $p(z)$, we employ the standard Gaussian distribution $\mathcal{N}(0, I)$.

In this study, we employ Optimus, a large-scale language VAE model as our baseline model (Li et al., 2020). Optimus is a Transformer-based VAE model that incorporates BERT (Devlin et al., 2019) as its encoder and GPT-2 (Radford et al., 2019) as its decoder, achieving higher performance among VAE language models. In the Optimus framework, the functionalities of BERT and GPT-2 are integrated as follows: within the BERT structure, the first token of each sentence is assigned as a unique classification token, marked by `[CLS]`. The final layer's hidden state, $\mathbf{h}_{\text{[CLS]}} \in \mathbb{R}^H$ acts as the sentence-level representation. This representation is subsequently converted into a latent space, forming the latent variable $\mathbf{z} = W_E \mathbf{h}_{\text{[CLS]}}$. Here, $\mathbf{z}$ is a P-dimensional vector in $\mathbb{R}^P$, and W_E is the transformation matrix in $\mathbb{R}^{P \times H}$. The latent variable z is then integrated into the GPT-2 model to facilitate its decoding function.

To tackle the successful style transfer task, we propose the Successful Style Transferred Variational Autoencoder (SST-VAE). An overview of the model is depicted in Figure 6. The SST-VAE is based on on Optimus framework. The loss functions of SST-VAE are shown as follows.

$$\mathcal{L}_{SST} = \mathcal{L}_E + \alpha \mathcal{L}_{success} + \beta \mathcal{L}_{inverse} \quad (4)$$

where $\mathcal{L}_E$ represents the reconstruction loss, $\mathcal{L}_{success}$ denotes the success prediction loss, $\mathcal{L}_{inverse}$ refers to the inverse success prediction loss, and α and β are the weights assigned to these losses.

In SST-VAE, we freeze and retain the parameters of the encoder and decoder from fine-tuned Optimus, utilizing them in their existing states. For the latent variables, we add a single-layer neural network, enabling the latent variables to incorporate information related to the success label as follows.

$$\mathbf{z}' = W_A \mathbf{z} \quad (5)$$

where W_A is the transformation matrix in $\mathbb{R}^{P \times P}$.

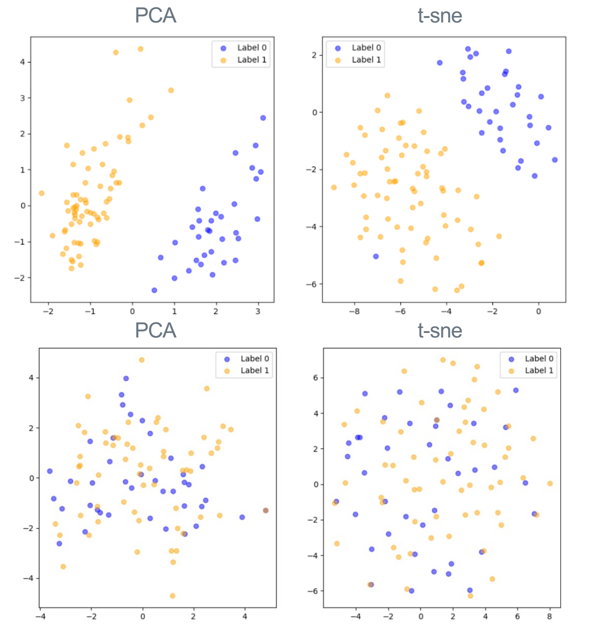

Figure 7: 2D Latent Embedding Plot: Top Row Shows SST-VAE, Bottom Row Baseline Models (Finetuned Optimus). Left Plots Utilize Principal Components, Right Plots Employ t-SNE for Dimension Reduction. The Yellow Labels Denote Successful Products, Blue Labels Indicate Non-Successful Products.

The success prediction loss is derived as follows.

$$\hat{\mathbf{x}} = Dec(\mathbf{x}, \mathbf{z}') \tag{6}$$

$$\mathbf{h}_{\hat{\mathbf{x}}} = BERT(\hat{\mathbf{x}}) \tag{7}$$

$$\hat{s} = f_\omega(\mathbf{h}_{\hat{\mathbf{x}}}) \tag{8}$$

$$\mathcal{L}_{success} = -s\log(\hat{s}) - (1-s)\log(1-\hat{s}) \tag{9}$$

First, we acquire the reconstructed text $\hat{\mathbf{x}}$ from the original text $\mathbf{x}$ and latent variables $\mathbf{z}'$ using the frozen decoder Dec. Then, we acquire the embeddings $\mathbf{h}_{\hat{\mathbf{x}}}$ of the reconstructed text with frozen BERT. Subsequently, we use a single-layer neural network f_ω to obtain the predicted success label $\hat{s}$. Finally, we calculate the success prediction loss with a cross-entropy loss with the success label $s \in \{0, 1\}$.

The inverse success prediction loss is derived as follows.

$$\mathbf{l}' = (1-2s)\mathbf{l} \tag{10}$$

$$\mathbf{x}' = Dec(\mathbf{x}, \mathbf{z}' + \mathbf{l}') \tag{11}$$

$$s' = f_\omega(\mathbf{x}') \tag{12}$$

$$\mathcal{L}_{inverse} = -s\log(s') - (1-s)\log(1-s') \tag{13}$$

where $\mathbf{l}$ is a learnable vector that represents the direction of success within the latent space, and $\mathbf{l}'$ is a vector that represents the opposite direction of success from the original success label s. The term $(1-2s)$ assigns a sign based on the success label $s \in \{0, 1\}$. When s is 1, $\mathbf{l}'$ becomes $-\mathbf{l}$ and when s is 0, $\mathbf{l}'$ becomes $\mathbf{l}$, reversing the sign of the embeddings. Then, we derive the style transferred text $\mathbf{x}'$ from the original text and the value of $\mathbf{z}'$ after adding $\mathbf{l}$. This process is to shift the latent variables in the direction opposite to the given success label, thereby generating descriptions with the inverse success label. Then, we utilize f_ω to predict the inverse success label. Finally, we obtain the inverse success prediction loss with a cross-entropy loss.

During inference, we can either add or subtract label embeddings to produce text with either more successful or less successful descriptions by $Dec(\mathbf{x}, \mathbf{z}' \pm \mathbf{l})$.

4.3. Evaluation Settings

For data processing, similar to the success prediction task, we eliminate products with less than 5 ratings and those with duplicated descriptions. Additionally, we eliminate product descriptions exceeding 64 tokens, in line with the preprocessing procedures in Optimus (Li et al., 2020). For data splitting, we utilize a stratified group shuffle method with a threshold set at 100.

As our baseline model, we fine-tune Optimus with a latent size of 768 and beta set to 1.0 [2], using our training corpus. Subsequently, the SST-VAE model is trained with the frozen weights of the encoder and decoder from the fine-tuned baseline model.

We assess our models' effectiveness in two approaches. Initially, we compare the latent embeddings of our model with those of the baseline model by projecting these embeddings into a 2D space through dimensionality reduction techniques such as principle component analysis and t-sne (van der Maaten and Hinton, 2008). Our second method involves qualitative evaluations through case studies. We generate style-transferred texts using both SST-VAE and a baseline model, then analyze the outcomes based on two viewpoints: assessing whether the results highlight factors behind product success and whether the transformed text enhances the original descriptions.

In our experiments, we chose hyperparameters as follows: The number of training epochs was set to 10. The loss weights were adjusted, with a ratio of 1/100 assigned to the reconstruction loss and a weight of 1 for the other two loss components. All models were trained on an NVIDIA V100 GPU.

[2] https://github.com/ChunyuanLI/Optimus/blob/master/doc/optimus_finetune_language_models.md

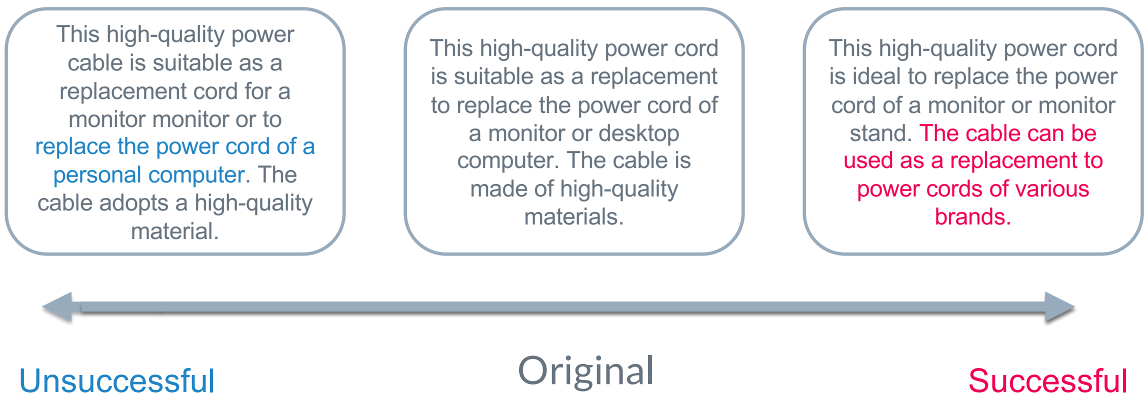

Figure 8: Example Output from SST-VAE Demonstrating Successful Style Transfer Task.

Style	Description
Original	Built-in Quick Operation Acoustic Engine for Sync and Phone Call Connectivity Advanced TX-Link for Listening to Music Auto-tuning LED-lit LCD TV 3.5mm Auxiliary Input for **iPod** Touch screen White Color with Built-in Dual Playback Jack PS 5 Input for Auxiliary Buit-in USB-C Audio Input Standard **Apple iPod** Touch 5intrusion 8x.
Unsuccessful	This high-quality power Opening-out Timer for Handsfree **Android** Auto Built-in Bluetooth 5.0 Microphone Input 3 Built-in USB Male Compatible with ALL Digital TV for Visual Pronotomic and Other Equinox RCA Light-Up Function iPod Controls 4 Touch for Enhanced Control of Control Skip-Forward/Forward Automatic Take-Down Timer .

Table 3: 1st Pattern: Describing Different Products.

4.4. Result and Discussion

The results of the embedding comparisons are presented in Figure 7. As Figure 7 demonstrates, the SST-VAE model exhibits clearer distinctions in the embedding space, suggesting that the latent embeddings from our proposed model provide more informative insights regarding the success of the product.

Figure 8 illustrates an example of the output generated by our model when it performs style transfer tasks. For a successful output, our model adds the successful label embeddings $\mathbf{l}$ to the latent variables $\mathbf{z}'$. Conversely, an unsuccessful output is generated when the model subtracts these label embeddings. Figure 8 presents the example of a power cord. When our model applies a successful style transfer, the description evolves to indicate the cord's compatibility with various brands, thereby suggesting its versatility. In contrast, an unsuccessful style transfer alters the description to specify the cord's exclusive use with personal computers, implying a more restricted utility.

Through qualitative analysis, we identified three distinct patterns in the output. The first pattern observed is a tendency to describe different products, as exemplified in Table 3. In this example, while the original text mentions Apple products like iPods, the descriptions shift to discussing Android when moved in the direction of unsuccessful styling. For automated product description generation, this deviation is not ideal, as the focus should remain on the specific features of the target product. Conversely, this tendency can be beneficial for understanding which brands or features might be more or less successful. This aspect is a unique point in extracting insights from product descriptions, as opposed to the automated generation where factual accuracy is paramount (Chan et al., 2019).

The second pattern we observed is that the model occasionally does not alter the output at all, even when we adjust the descriptions towards more successful or unsuccessful directions, as illustrated in Table 4. As the example indicates, this is problematic from both perspectives, as it neither facilitates automation nor provides insights into product success. We attribute this issue to the dominance of reconstruction loss in the loss functions. Upon examining the magnitude of the losses, we found that the reconstruction loss significantly outweighs others, such as success prediction and inverse success prediction losses. This remains true even when applying a weight of 0.01 to the reconstruction loss. On the other hand, using even

Style	Description
Original	You MUST RE-USE: your existing cabling and hardware. It is your responsibility to verify the batteries being ordered match the batteries in your unit prior to placing your order. Our products are not affiliated with or authorized by APC.
Successful	You MUST RE-USE: your existing cabling and hardware. It is your responsibility to verify the batteries being ordered match the batteries in your unit prior to placing your order. Our products are not affiliated with or authorized by APC.
Unsuccessful	You MUST RE-USE: your existing cabling and hardware. It is your responsibility to verify the batteries being ordered match the batteries in your unit prior to placing your order. Our products are not affiliated with or authorized by APC.

Table 4: 2nd Pattern: Generating the Same Descriptions.

Style	Description
Original	Higher capacity will takes **3-4 hours** to fully charge the battery.
Successful	High capacity can replace the battery to charge large volumes of batteries **quickly**.
Unsuccessful	High-capacity battery will takes **more time to fully charge**.

Table 5: 3rd Pattern: Hallucinations.

smaller weights for the reconstruction loss resulted in outputs that were grammatically incorrect and not viable as sentences. Therefore, balancing the generation of valid sentences with the capability to modify texts towards more successful or unsuccessful versions presents a key area for future research.

The third observed pattern is a tendency towards hallucinations, as shown in Table 5. In the provided example, the original text states that charging the battery takes 3 or 4 hours. However, when we transform the text into a more successful description, it incorrectly claims that the charge will finish quickly. This pattern, akin to the first one, poses challenges for automating product description writing. On one hand, the features emerging in the successful direction could be leveraged for product improvement.

5. Conclusion

In conclusion, this study contributes to the evolving field of e-commerce NLP by taking initial steps in extracting insights about product success from product descriptions.

Our research addressed two research questions: “To what extent can product descriptions accurately indicate a product’s likelihood of success?” and “Can we transform text descriptions from unsuccessful to successful ones, or vice versa?”

Our findings for the first research question revealed that product descriptions were indeed good predictors of the products’ success with an accuracy of around 70% in success prediction tasks, showing that they contain meaningful information about the product’s success. On one hand, the exact mechanisms behind this successful prediction remain an area for future exploration.

Regarding the second research question, we proposed a Successful Style Transfer Variational Autoencoder (SST-VAE). Our findings indicated that the proposed model successfully transforms the product descriptions from unsuccessful to successful and vice versa through qualitative analysis and case studies. Also, we showed that the proposed model is effective in extracting insights about products by highlighting the successful factors. On one hand, qualitative analysis revealed challenges such as hallucinations and the balance between factual accuracy and style transformation in the generated text. These findings pave the way for future research focused on addressing these challenges to progress the domain of e-commerce NLP.

6. Bibliographical References

Zhangming Chan, Xiuying Chen, Yongliang Wang, Juntao Li, Zhiqiang Zhang, Kun Gai, Dongyan Zhao, and Rui Yan. 2019. Stick to the facts: Learning towards a fidelity-oriented E-commerce product description generation. In *Proceed-*

ings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4959–4968, Hong Kong, China. Association for Computational Linguistics.

Qibin Chen, Junyang Lin, Yichang Zhang, Hongxia Yang, Jingren Zhou, and Jie Tang. 2019. Towards knowledge-based personalized product description generation in e-commerce. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pages 3040—-3050, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *International conference on machine learning*, pages 1587–1596. PMLR.

Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. 2022. Deep learning for text style transfer: A survey. *Computational Linguistics*, 48(1):155–205.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. Disentangled representation learning for non-parallel text style transfer. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 424–434, Florence, Italy. Association for Computational Linguistics.

Diederik P Kingma and Max Welling. 2014. Auto-encoding variational bayes. *ICLR*.

Chunyuan Li, Xiang Gao, Yuan Li, Baolin Peng, Xiujun Li, Yizhe Zhang, and Jianfeng Gao. 2020. Optimus: Organizing sentences via pre-trained modeling of a latent space. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4678–4699, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Jake Vasilakes, Chrysoula Zerva, Makoto Miwa, and Sophia Ananiadou. 2022. Learning disentangled representations of negation and uncertainty. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8380–8397, Dublin, Ireland. Association for Computational Linguistics.

Jinpeng Wang, Yutai Hou, Jing Liu, Yunbo Cao, and Chin-Yew Lin. 2017. A statistical framework for product description generation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 187–192, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Tao Zhang, Jin Zhang, Chengfu Huo, and Weijun Ren. 2019. Automatic generation of pattern-controlled product description in e-commerce. In *The World Wide Web Conference*, WWW '19, pages 2355—-2365, New York, NY, USA. Association for Computing Machinery.

Guineng Zheng, Subhabrata Mukherjee, Xin Luna Dong, and Feifei Li. 2018. Opentag: Open attribute value extraction from product profiles. KDD '18, pages 1049—-1058, New York, NY, USA. Association for Computing Machinery.