Attribute-Sentiment-Guided Summarization of User Opinions From Online Reviews

Yi Han
Department of Mechanical
and Industrial Engineering,
Northeastern University,
Boston, MA 02115
e-mail: han.yi1@northeastern.edu

Gaurav Nanda
School of Engineering Technology,
Purdue University,
West Lafayette, IN 47907
e-mail: gnanda@purdue.edu

Mohsen Moghaddam[1]
Department of Mechanical and Industrial
Engineering and Khoury
College of Computer Sciences,
Northeastern University,
Boston, MA 02115
e-mail: mohsen@northeastern.edu

Eliciting informative user opinions from online reviews is a key success factor for innovative product design and development. The unstructured, noisy, and verbose nature of user reviews, however, often complicate large-scale need finding in a format useful for designers without losing important information. Recent advances in abstractive text summarization have created the opportunity to systematically generate opinion summaries from online reviews to inform the early stages of product design and development. However, two knowledge gaps hinder the applicability of opinion summarization methods in practice. First, there is a lack of formal mechanisms to guide the generative process with respect to different categories of product attributes and user sentiments. Second, the annotated training datasets needed for supervised training of abstractive summarization models are often difficult and costly to create. This article addresses these gaps by (1) devising an efficient computational framework for abstractive opinion summarization guided by specific product attributes and sentiment polarities, and (2) automatically generating a synthetic training dataset that captures various degrees of granularity and polarity. A hierarchical multi-instance attribute-sentiment inference model is developed for assembling a high-quality synthetic dataset, which is utilized to fine-tune a pretrained language model for abstractive summary generation. Numerical experiments conducted on a large dataset scraped from three major e-Commerce retail stores for apparel and footwear products indicate the performance, feasibility, and potentials of the developed framework. Several directions are provided for future exploration in the area of automated opinion summarization for user-centered design. [DOI: 10.1115/1.4055736]

Keywords: user-centered design, text summarization, natural language processing, multi-instance inference, language model, text-to-text transfer transformer (T5), artificial intelligence, machine learning, product development

1 Introduction

Understanding user needs is a preliminary step in early-stage product development [1]. User feedback plays a key role in product design and development, as it provides important information about user interaction experiences with various attributes of a product to designers and manufacturers. The increasing use of e-Commerce platforms has resulted in large and rich collections of user feedback in the form of online product reviews [2]. One of the main advantages of analyzing online reviews is that they contain detailed and nuanced feedback from large and diverse user populations on different attributes of various competing products [3,4], which may not be the case in pilot launches, small-scale usability studies, or focus groups involving product design and development teams [5–7]. However, it is challenging to comprehend a large collection of textual reviews that typically address the varied user experiences and sentiments associated with different attributes of a product. The question thus remains on how to accurately elicit user needs from online reviews at scale and relay that information to designers in a useful format.

Natural language processing (NLP) techniques such as text summarization, sentiment analysis, and topic modeling can be utilized to extract important themes from a collection of user reviews [8,9]. Among these, models require qualitative interpretation of generated topics, which often requires significant effort and time. Sentiment analysis has been widely explored in the user need finding literature [10–18]. However, the results of the sentiment analysis process, which often takes the form of sentiment polarity or intensity values, may inherently lose important information that could otherwise be useful to designers. Text summarization approaches can address this shortcoming by providing compiled summaries of important points covered in a large collection of reviews, which can be used directly by product designers for further analysis [17]. There are mainly two types of text summarization approaches: extractive [19–21] and abstractive [22–24]. The former extracts and concatenates key sentences or paragraphs from the original text without necessarily capturing their context or meaning, while the latter leverages language models to generate text in a more advanced fashion, similar to human interpretation.

1.1 Knowledge Gaps.

The existing need finding approaches are based primarily on qualitative analysis of previous designs, surveys, or focus group studies, which are inherently biased due to the targeting of a small fraction of users and product instances with structured inquiries. The growing abundance of user feedback data in the form of online reviews, tweets, comments, or forum discussions has created new opportunities for designers and product developers to elicit user needs at scale. Sentiment analysis has been a key enabler for large-scale need finding from user-generated data over the past decade [25–28]. However, the current research is mainly focused on sentiment classification at the attribute, sentence, or document level [29–36], which would inevitably lead to information loss due to the aggregation and quantification of user feedback and opinions.

Text summarization is another NLP technique explored in the literature to extract user needs in the form of opinion summaries [37–44]. Yet, most of the existing opinion summarization approaches are extractive in nature and place emphasis on the

[1] Corresponding author.

Contributed by the Design Theory and Methodology Committee of ASME for publication in the Journal of Mechanical Design. Manuscript received May 12, 2022; final manuscript received September 10, 2022; published online December 9, 2022. Assoc. Editor: Christopher McComb.

percentage of information that could be extracted, which could result in potential information loss in the text summarization process due to the disregard of contradictory opinions in different reviews. To be more specific, most existing research merely evaluates the quality of the summarization results with respect to the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) score [45], which essentially counts the number of overlapping units such as n-grams [46], word sequences, and word pairs between computer-generated summaries and ground-truth summaries created by humans. However, the ROUGE score could not provide enough support to assess the "direction" of the summary with respect to the attributes discussed or the general sentiment of the summary.

In multidocument summarization tasks [47,48] such as summarizing multiple user reviews (i.e., documents), different documents may contain totally contradictory opinions. Therefore, the generated summary can be easily influenced by the "dominant" opinions, i.e., the opinions with a greater representation in the corpus in terms of the lengths of reviews and/or the number of reviews with the same opinion. This bias in summarization occurs because the ROUGE score encourages the generated summary to contain more text from the longer text [49,50]. This peculiarity can undermine the ability of the summarization process to generate informative and representative summaries. This article aims to bridge this gap by "steering" the summarization process using the attribute-level sentiments of users extracted from the review set. Such a controlled summarization [51] would allow designers to generate attribute-specific summaries of user reviews with similar sentiments.

1.2 Objectives. This article builds and validates a hierarchical Multi-instance Attribute-Sentiment (MAS) inference model to infer attributes and sentiments from each individual review as well as each sentence and word inside the review. Using a well-trained MAS inference model, a synthetic dataset is assembled in the form of attribute-specific summaries with sentiment labels. The synthetic dataset is then used to fine-tune a pretrained language model, text-to-text transfer transformer (T5) model [52], to generate abstractive summaries. The general framework shown in Fig. 1 has the ability to generate abstractive summaries from a raw review corpus guided by specific attributes of product and sentiment preferences. The main contributions of this article are as follows:

- A new synthetic data set is created that can be used for both attribute-level sentiment analysis and attribute-sentiment-guided summarization of user needs from online reviews. The dataset includes both raw reviews from several online platforms and (reviews, summary) pairs that could serve the training purpose of text summarization
- A multimodel computational framework is built, which includes (a) sentiment-attribute information integrated by the MAS inference model and (b) a fine-tuned T5 model trained using the results from the MAS inference model for generating summaries with specific attributes and sentiments.
- The proposed end-to-end framework for inferring attribute-sentiment-specific summaries of user opinions is tested and validated through experiments on a large dataset of user reviews scraped from multiple e-Commerce platforms for footwear. The experiments also illuminate the impact of attribute-sentiment categorization on the quality of generated opinion summaries.

The remainder of this article is organized as follows. Section 2 provides a summary of background work related to the main research topics, including text summarization and sentiment analysis.

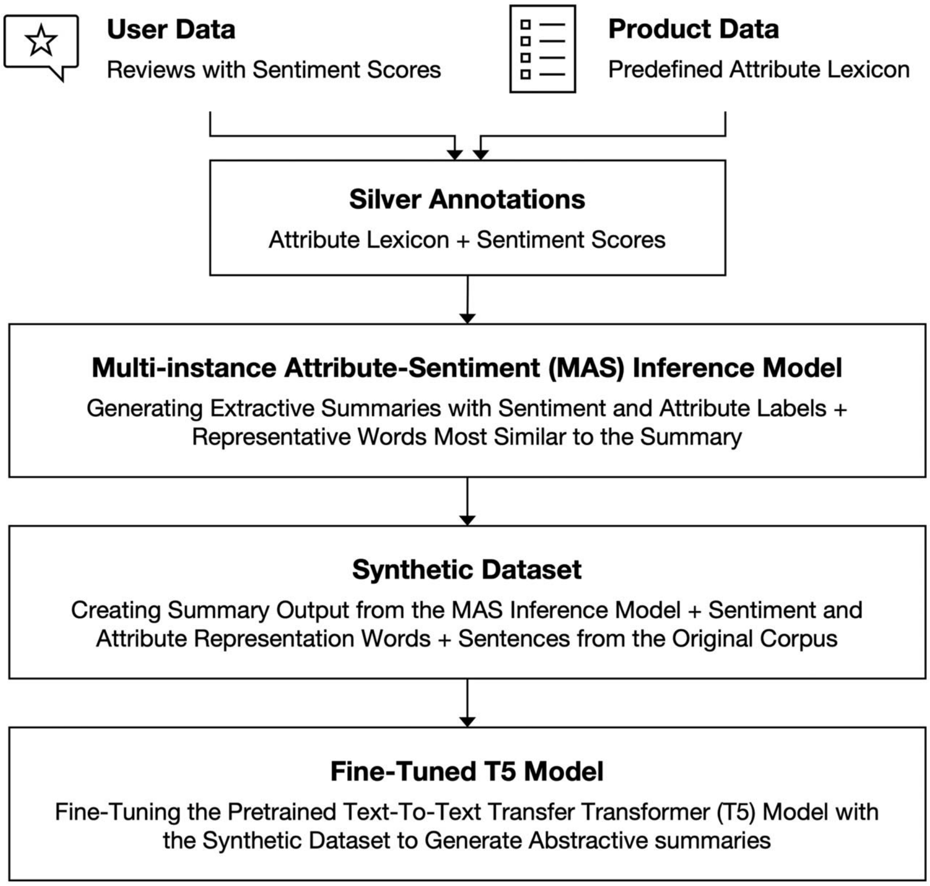
User Data
Reviews with Sentiment Scores
Product Data
Predefined Attribute Lexicon
Silver Annotations
Attribute Lexicon + Sentiment Scores
Multi-instance Attribute-Sentiment (MAS) Inference Model
Generating Extractive Summaries with Sentiment and Attribute Labels + Representative Words Most Similar to the Summary
Synthetic Dataset
Creating Summary Output from the MAS Inference Model + Sentiment and Attribute Representation Words + Sentences from the Original Corpus
Fine-Tuned T5 Model
Fine-Tuning the Pretrained Text-To-Text Transfer Transformer (T5) Model with the Synthetic Dataset to Generate Abstractive summaries

Fig. 1 The proposed hierarchical MAS inference model for attribute-sentiment-guided opinion summarization

Section 3 discusses the details of the proposed framework, including the MAS inference model, the synthetic dataset creation process, and the T5 model fine-tuning process. Section 4 presents the experimental results, analyses, and implications of the developed methodology. Section 5 provides concluding remarks and several directions for future research.

2 Background

This section presents a brief overview of the state-of-the-art in text summarization and attribute-level sentiment analysis, along with their implications for user need finding in early-stage product development processes.

2.1 Text Summarization Using Synthetic Data.

The previous work on text summarization focuses mainly on general opinion summarization using abstractive methods [22–24] or extractive methods [19–21]. Since annotated opinion summary datasets for training are rare and difficult to generate on a large scale, most recent studies approach opinion summarization as an unsupervised learning problem for which only the review corpus is available [44,53,54]. State-of-the-art unsupervised summarization approaches utilize autoencoders [55] to first train a text decoder by reconstruction and use it to generate summaries based on inputs. Considering the unsupervised nature of these methods, the quality of their generated summaries is often much lower compared to the summaries generated by supervised methods.

Some recent studies have proposed abstractive summarization models that can generate overall and attribute-specific summaries and have evaluated their performance in user reviews on services such as hotels and restaurants [56,57]. A key limitation of guiding the summaries only with respect to attributes is mixing up contradictory or positive/negative opinions of users about particular attributes. To address these gaps, this article develops a framework for abstractive summarization of user reviews to generate attribute-specific and sentiment-specific summaries. The performance of the model is demonstrated and evaluated on a large review dataset of sneakers scraped from multiple e-Commerce platforms.

For sentiment analysis, polarity and subjectivity are considered as two main dimensions and are determined for various attributes of the product as well as for the overall product. The sentiment polarity score indicates the intensity of emotions expressed by the user, for example, extremely negative/unhappy, neutral, moderately positive, or highly positive. The sentiment subjectivity score indicates whether the review was largely a subjective opinion, for example, “I did not like the shoe sole” or was objective in nature, for example, “The shoe sole was very narrow.” Both these sentiment dimensions contain important information about user experience, which is mostly complementary in nature. As this is a pilot study, the sentiment intensity at the attribute level was used to group the data into unique combinations of attribute and sentiment polarity, and the reviews were summarized for each group using an abstractive summarization approach. The findings of this study would be useful to researchers in the engineering design domain, as well as product designers and manufacturers.

Abstractive summarization based on synthetic data has been proven feasible in the past. Synthetic datasets are usually generated through unsupervised methods such as autoencoder [58], noising, and denoising [59], ranking good reviews by similarity and reusing them as summary [60], or through supervised methods that utilize attribute controller systems [61]. The latter is based on the idea of assembling pairs (reviews, summary) from a review corpus as synthetic data to train a supervised learning model. This article uses the supervised learning approach to generate the synthetic dataset.

2.2 Summarization Guided by Attribute-Level Sentiment Analysis.

The main goal of any text summarization task is to capture as much critical information as possible with the minimum number of words [62]. This could be achieved by guiding the summary generation process with respect to specific keywords (e.g., attributes/aspects of a product or a service) as the main subjects of the summary [63,64]. The summarization process can also be guided by user sentiments, for example, to compare the generated summaries of positive, negative, and neutral opinions [11–13,65,66]. Although both text summarization and sentiment analysis are popular research topics in the NLP domain, studies that incorporate both as complementary capabilities are rare. For example, some studies have attempted to build an efficient text summarization system by selecting sentiment keywords [67], which could roughly provide the direction of the summarization. Other studies have “concatenated” the two techniques by first precategorizing the corpus with respect to different sentiment values and then choosing keywords to guide the summarization process [68–70]. Some studies have used sentiment analysis as an independent, prefilter to improve the reliability of the summarization task [71]. To our knowledge, no existing study has explored the possibility of integrating the opinion summarization and sentiment analysis processes in a single model to enable controllable generation of summaries with respect to user-specified attributes and sentiments.

3 Methodology

This section presents the four main steps of the overarching framework for attribute-sentiment-guided summarization (see Fig. 2), including data collection and preprocessing, building the MAS inference model, creating the synthetic dataset, and fine-tuning the T5 model for abstractive summarization. The overall flow of the model with the output of each stage is presented in Fig. 2.

3.1 Collecting and Preprocessing the Data.

The review corpus used in this article is scraped from multiple online apparel and footwear stores including Finish Line[2], New Balance[3], and Asics[4]. This corpus contains over 140K reviews of sneakers. However, not all of the reviews are informative and useful for analyzing the user needs. A review filtering procedure is therefore designed as explained in Sec. 4. The corpus comes with a document sentiment label represented by a range of stars from 1 to 5. In addition to the overall rating, this article assigns an objectivity label to each review as well as an attribute label based on a prespecified attribute lexicon [18], which comprises over 200 attribute words related to footwear grouped in seven categories including permeability, impact absorption, stability, durability, shoe parts, exterior, and fit. The detailed lexicon can be found in Sec. 4. Since this work involves two types of labels, the balance of the labeled dataset should be considered with respect to both types of labels.

Since MAS model only requires two types of labels, including attributes and sentiments, the labeling process is efficient with the review stars and the attribute lexicon. Without loss of generality, the sentiment label was assigned by the customer star: [positive: 5 stars, neutral 3-4 stars, negative: 1-2 stars], the attribute label was assigned with the attribute lexicon by word matching. The reason for choosing the proposed rating scale was to make the three categories more balanced. Even four-star reviews were found to generally complain about some aspects of the product; therefore, reviews with fewer than five stars are not completely positive. In addition, this rating scale does not affect the model architecture and training process and can be modified in other application domains where four-star reviews are more positive. Further, the proposed rating scale helps balance the dataset, because 75.89% of the reviews in the dataset are five stars, while less than 5% reviews are three stars; hence, the size of the neutral category must be expanded.

3.2 Building the Multi-Instance Attribute-Sentiment Inference Model.

The MAS inference model is a supervised

[2] www.finishline.com
[3] www.newbalance.com
[4] www.asics.com

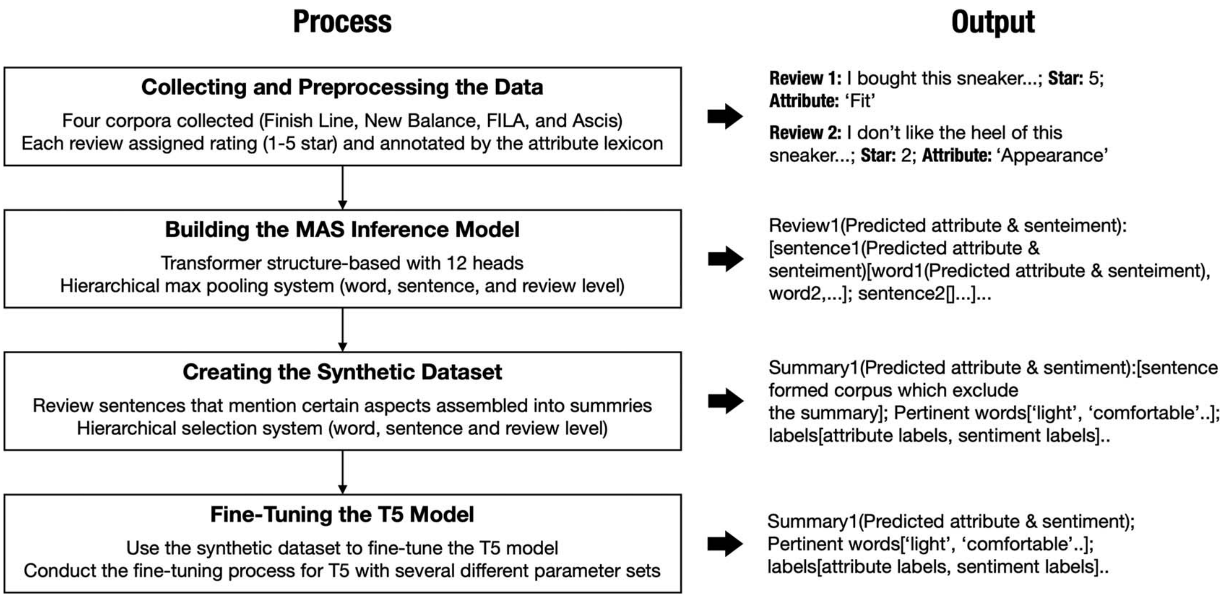
Process
Output
Collecting and Preprocessing the Data
Four corpora collected (Finish Line, New Balance, FILA, and Ascis)
Each review assigned rating (1-5 star) and annotated by the attribute lexicon
Review 1: I bought this sneaker...; Star: 5; Attribute: 'Fit'
Review 2: I don't like the heel of this sneaker...; Star: 2; Attribute: 'Appearance'
Building the MAS Inference Model
Transformer structure-based with 12 heads
Hierarchical max pooling system (word, sentence, and review level)
Review1(Predicted attribute & senteiment): [sentence1(Predicted attribute & senteiment)[word1(Predicted attribute & senteiment), word2,...]; sentence2[]...]...
Creating the Synthetic Dataset
Review sentences that mention certain aspects assembled into summries
Hierarchical selection system (word, sentence and review level)
Summary1(Predicted attribute & sentiment):[sentence formed corpus which exclude the summary]; Pertinent words['light', 'comfortable'..]; labels[attribute labels, sentiment labels]..
Fine-Tuning the T5 Model
Use the synthetic dataset to fine-tune the T5 model
Conduct the fine-tuning process for T5 with several different parameter sets
Summary1(Predicted attribute & sentiment); Pertinent words['light', 'comfortable'..]; labels[attribute labels, sentiment labels]..

Fig. 2 The overall model flow and every step output for attribute-sentiment-guided opinion summarization

machine learning framework in which the labels correspond to a bag of instances that have not been labeled [72]. The goal of the model is to identify the bag labels of those unlabeled instances. In this article, a hierarchical model structure is designed to predict review labels from sentence and word predictions. The rationale behind choosing the multi-instance model is the similarity of the model structure to the process used by humans to generate summaries. Thus, in the data labeling process, the first step is to filter useful sentences from a set of reviews. When creating the attribute-related summary, the annotators generate content from sentences related to the attribute of interest. They then summarize those sentences into a single summary, which must incorporate the same label as those sentences. In the MAS inference model, the sentiment label is added along with the attribute label with three types of polarities associated with the review: positive, neutral, and negative. In this case, the stars provided for reviews on the e-Commerce platform are used to induce user sentiments. Although language models such as BERT [73] can provide embeddings at token, sentence, and document levels for classification, they cannot handle the desired hierarchical voting mechanism, as they require datasets with sentence-level and word-level labels that may not be necessarily available. Therefore, this article applies the MAS model to generate these labels, as described in the remainder of this section.

3.2.1 Model Structure. To develop an abstractive summarization model through supervised learning, a labeled dataset is required that includes review-summary pairs. However, such datasets are rare and hard to generate. In such cases, several studies have attempted to train supervised learning models by creating synthetic datasets, which have shown remarkable performance [58–60]. Building on this idea, this article conducts abstractive opinion summarization through a three-stage process: (1) train the MAS model with a review-based dataset (Fig. 3), (2) generate synthetic dataset with the output of the MAS model, and (3) fine-tune T5, a state-of-the-art sequence-to-sequence model, using the synthetic dataset to generate abstractive summaries for specified attributes and sentiment polarities. The overall structure of the proposed

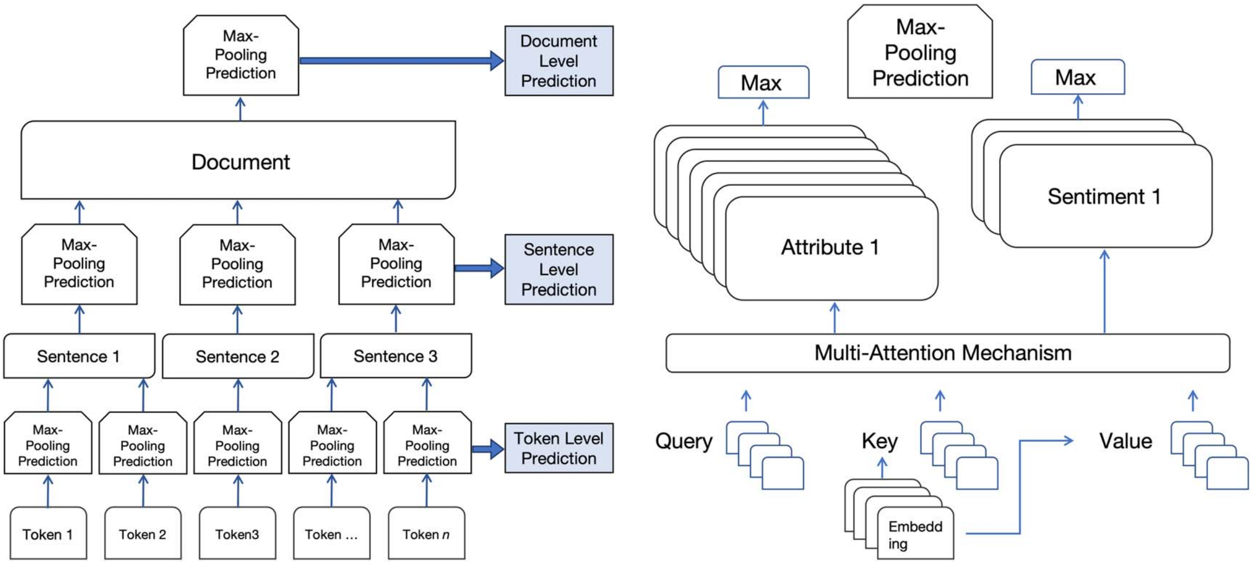

Fig. 3 The MAS model

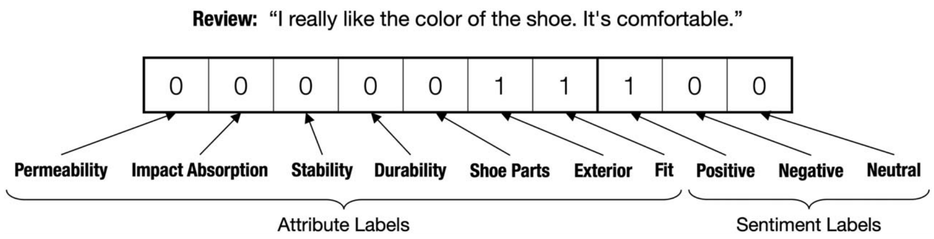

Fig. 4 Example of the MAS model output

model is similar to the attribute-controllable summarization model, AceSum [61], with the following additional features:

- AceSum only provides an attribute controller, while the proposed model also incorporates sentiment polarities via the sentiment controller. Further, the AceSum model may yield no output, yet the proposed model always predicts at least one label, which is the sentiment.
- The multi-instance model of AceSum creates the synthetic dataset using less than ten seed words, while the proposed model generates the synthetic dataset using a rich attribute lexicon previously developed by the authors [18].
- AceSum uses a soft-margin (SM) loss function for the multi-instance model because their label set was binary with -1 and 1. The proposed model, however, uses the sigmoid binary cross-entropy (BCE) loss function for training to reduce the influence of the unbalanced dataset with respect to attributes and sentiments.
- In the synthetic data creation process, AceSum assumes that in a review set, each review that fulfills some constraints is a summary, and all the rest is the training corpus. The proposed model, however, does not use one entire review as a summary, but rather ranks all the sentences from the reviews in the corpus with respect to their relevance to the desired attributes and sentiment polarities and assembles the top-ranked sentences as a summary.

3.2.2 Model Formulation. To generate the synthetic training dataset for the downstream summarization task, the MAS model is designed to generate two types of labels at three levels: attribute labels and sentiment labels. The MAS model generates attribute labels and sentiment labels at the word level, sentence level, and review level. For example, given seven attribute categories "Permeability," "Impact Absorption," "Stability," "Durability," "Shoe Parts," "Exterior," and "Fit," three sentiment categories "positive," "negative," and "neutral," a user review "I really like the color of the shoe, it's comfortable," the MAS model will generate the output shown in Fig. 4.

The MAS model is formulated as follows (Fig. 3). Let C denote the corpus that includes reviews with one to five stars provided by users, and $A = a_1, a_2, \ldots, a_n$ denote the set of attributes [18]. Each review r_i can be formulated as a list of words $w_1, w_2, \ldots, w_n$. For a given review with word list W_n, the RoBERTa (Robustly Optimized Bidirectional Encoder Representations from Transformers Approach) tokenizer [49] RB is utilized for encoding, expressed as $e = RB(W_n)$. The proposed model uses the label $\{0, 1\}$ to indicate the labels (e.g., see Fig. 4). Thus, the token-level prediction P_t can be obtained using a nonlinear transformation:

$$P_t = \text{ReLU}(W_e + b) \tag{1}$$

where b is a constant added to the linear and nonlinear transformation of the word encoding, and ReLU denotes the Rectified Linear Unit.

The model then uses token-level predictions to induct sentence-level predictions P_s. The induction process uses the multiple attention mechanism [74] (Fig. 3). Through the max pooling process, each word in the sentence votes for the most pertinent attribute and sentiment label, and each sentence in the review votes for the same label. The proposed model uses 12 attention heads. That is, the batched result of P_t is split into 12 heads, denoted by h. Each key key_h is transformed using a nonlinear transformation. To allow better differentiation, tanh and Relu are examined during training, where ReLU indicated better performance. Thus, ReLU activation functions are used in the attention mechanism of the MAS model, as follows:

$$key_h = \text{ReLU}(W_{he} + b_h) \tag{2}$$

Other settings for the attention mechanism follow the original AceSum model. Each attention output is calculated as follows:

$$a_h = \text{softmax}(key_h \cdot query_h) \tag{3}$$

The head attention prediction P_h is calculated as $P_h = \sum_k (p_t a_h[k])$. Each attention head in the model represents a semantic space of the review. Thus, sentence-level predictions P_{sa} are calculated as follows:

$$P_{sa} = \text{maxpooling}(P_h) \tag{4}$$

Similarly, the predictions for sentiments at the sentence-level P_{ss} are calculated as follows:

$$P_{ss} = \text{maxpooling}(P_h) \tag{5}$$

In the same vein, the model uses sentence-level prediction to induce review-level prediction P_r.

3.2.3 Loss Function. The MAS model is intended to be used to create the synthetic dataset for summarization. Thus, the model must be highly capable of classification and information retention. The AceSum model uses a soft-margin loss function, which shows remarkable performance. Yet, the soft-margin loss function is not suitable for the proposed MAS model since there are two types of probability-based labels, that is, (0,1). In this article, several loss functions including soft-margin loss, multilabel margin loss, and cross-entropy were experimentally evaluated, and the weighted multiple binary cross-entropy loss function was selected due to providing the best performance. The comparison results are presented in Sec. 4. Accordingly, the following loss function is formulated for the MAS model:

$$\mathcal{L}_{\text{MAS}} = -w_{n,c}[\mathcal{P}_c \hat{y}_{n,c} \log \sigma(y_n) + (1 - \hat{y}_{n,c}) \log (1 - y_{n,c})] \tag{6}$$

where $w_{n,c}$ denotes the weights of each label category c in the MAS model, n is the batch size, y is the actual label, and $\hat{y}$ is the predicted label. The category weights $\mathcal{P}_c$ are calculated using the following function:

$$\mathcal{P}_c = \frac{\text{Negative sample number}}{\text{Positive sample number}} \tag{7}$$

3.3 Creating the Synthetic Dataset. To assemble the supervised learning dataset for the abstractive summarization task, the output of the MAS model is used. The synthetic dataset should be in the following format: [Summary:…; Reviews:…; Keyword:…; Aspect:…]. In the synthetic data creation process, no human-annotated data were used and the entire training corpus was assembled by the MAS output. Unlike existing approaches that simply choose one individual review from a set of reviews as a synthetic summary for training [61], this article develops a more effective approach to generate the synthetic dataset. Since the MAS model can provide sentiment and attribute predictions for each sentence in the corpus, the proposed approach chooses the top three reviews from the 60 reviews in each batch loop as a mini-corpus to assemble a summary. The three selected reviews must contain the same sentiment and address the attribute of interest. Since attribute labels are usually imbalanced (e.g., "Shoe Parts", "Exterior", and "Fit" are discussed more frequently), the probability of multiple labels appearing in the same summary is small. Choosing reviews that follow this protocol (i.e., candidate reviews with only one common attribute) is shown to lead to better results.

The proposed synthetic summary generation approach ensures that the assembled summary (a) has a consistent sentiment and (b) contains at least three sentences that explicitly mention an attribute. If multiple attributes appeared in one sentence, the sentence would still be selected. To formulate the synthetic dataset creation method in this work, consider each review r_i in the batch along with two labels P_s (i.e., sentence-level prediction) and P_a (i.e., attribute-level prediction). The synthetic summary generation process first goes through all the reviews in the batch and group all the reviews having the same sentiment, and then select at least three reviews containing an attribute of interest. That is, sentences with the same P_{ss} (i.e., sentiment predictions) and at least one common P_{sa} (i.e., sentence-level attribute predictions) are collected to assemble the synthetic summaries. The remaining sentences are then ranked by similarity, calculated using the soft-margin score as follows:

$$\text{Similarity} = \sum_{j} \frac{\log(1+\exp(-\hat{y}(j)y(j)))}{\text{length}(y)} \tag{8}$$

where $\hat{y}$ is the target, y is the input, and j is the batch sample. During the experiments, the upper bounds for the training corpus were set as 50 sentences and 500 words. This limit is the filter when assembling the training corpus for T5 to ensure the quality of the generated summaries. The upper limit for the summarization corpus was then set to 50 sentences and 500 words. In the synthetic data creation process, the Natural Language Toolkit [75] and Spacy [76] were used to remove the stop words from the keyword list output. The format of the synthetic data is as follows: {summary, reviews, keywords, attribute and sentiment labels}, as illustrated in the following example:

- *Summary*. "Love it. They're perfect! I have always worn Asics for running but I wear these even when I'm not exercising… The color combo makes them my favorite pair of Asics! My first pair of ASICS and I will never go back to Nike. Being able to pick a shoe that meets my foot needs is fantastic, it truly makes a difference in the comfort of my workouts. I have recently bought three pairs of the Zig Kinetica shoes. They are the most comfortable workout/casual shoe that I have ever worn! Due to foot surgery in Oct., many of my athletic shoes are not comfortable anymore. These shoes are particularly great for women who need more width in the toe box."
- *Reviews*. ["So comfortable!","They're super comfortable and warm!", "I would buy another pair!", "First shoe my 15 year old picked himself.", "Very comfortable!",…].
- *Keywords*. ["fit","comfy", "comfort", "exercising","black",…].
- *Attribute and sentiment labels*. [0, 0, 0, 0, 1, 0, 1, 1, 0, 0]. The first seven values are the attribute labels and the last three are the sentiment labels.

3.4 Fine-Tuning the T5 Model.

After creating the synthetic dataset, several state-of-the-art pretrained transformer-based models were examined and the T5 [52] was selected to generate abstractive summaries. T5 is shown to provide state-of-the-art performance in the abstractive text summarization task [77–79]. The AceSum paper uses the original T5 for summarization. In this article, a new version of T5 named T5 V1 was fine-tuned and tested. The performance of the T5 model has been compared with other alternative models, as discussed in Sec. 4. During the fine-tuning process, the outputs of the MAS model are assembled in the following format:

[attribute][attribute1][attribute2][attribute..][Sentiment] [KEY]
keyword1, keyword2, keyword3 … [SNT] sentences …

where [*attribute*] indicates the current summary related to a certain attribute, [*sentiment*] indicates the sentiment polarity of the current model, and [*KEY*] and [*SNT*] correspond to the selected keywords and sentences. P is used as input to produce the encoding E. The decoder then outputs a token distribution $p(y_t)$ conditioned on the T5 attention mechanism. In this way, the training corpus was concatenated into several independent pieces. Sentiment and attribute information was added in the last piece. Through this assembly mechanism, the T5 model can learn to guide the summarization process toward the intended attribute and sentiment labels. The T5 model is fine-tuned using a maximum likelihood function:

$$\mathcal{L}_{sum} = -\sum \hat{y}_n \log p(y_n) \tag{9}$$

where $\hat{y}$ is the target, y is the input, and n is the batch size. During the training process, the output of the model could be controlled by manipulating the attribute controller and the sentiment controller. Moreover, the model has the ability to induct the overall summary by selecting all attributes during training. Several special settings are required for training as follows:

- *Cosine scheduler*. During the training process, the cosine scheduler is used to create a schedule with a learning rate that decreases after the values of the cosine function drops from the initial learning rate (*lr*) in the optimizer to 0, with several hard restarts after a warm-up period during which it increases linearly from 0 and to the initial learning rate set in the optimizer. As a functional optimizer, constant scheduler and linear scheduler are also available in the library, and in this article, cosine scheduler is shown to provide the best performance.
- *Dropout design*. During training, dropout randomly zeroes some elements of the input tensor with the probability p, using samples from a Bernoulli distribution. Each channel is zeroed independently on each forward call. This has proven to be an effective technique for regularization and prevention of co-adaptation of neurons [80]. Furthermore, the results are scaled by a factor of $\frac{1}{1-p}$ during training. This means that during evaluation, the module simply computes an identity function.
- *Beam search for summary generation*. In the sequence-to-sequence model, there are two main parts to provide the summary: an encoder, which is in charge of transferring sentences into embeddings, and a decoder, which translates trained embeddings back into sentences. On the output side, summaries are generated word by word. Beam search offers the option to select multiple words at the same time. In this article, the best beam search size during the training process was identified as 2.

The output of the fine-tuned T5 model consists merely of review summaries, since keywords and labels are only used as input for the training process. By modifying the input switches, the fine-tuned T5 model is able to provide sentiment-attribute-guided summaries.

The F1 score is used to measure the performance of the MAS model, and ROUGE-L score is used to judge the T5 summarization model. The F1 score measures the balance of precision and recall to provide a more realistic measure of the model performance. It is measured as the weighted harmonic mean of precision and recall, as follows:

$$\mathrm{F1} = \left(\frac{\mathrm{precision}^{-1} + \mathrm{recall}^{-1}}{2}\right)^{-1} \tag{10}$$

ROUGE-L is based on the length of the longest common subsequence of candidates and references [81]. Further, subjectivity and polarity scores are used to analyze the summarization performance using the Textblob [82] package, which calculates the subjectivity score for a review by averaging the subjectivity score determined for each word using a Naive Bayes model trained on hand-labeled reference dataset.

4 Experiments and Results

The first step to implement the proposed MAS-T5 summarization framework is to clean and filter the corpus. To this end, the following steps must be taken:

- Remove reviews that are too short or too long. In this research, the lower limit of the length of useful review was set up as 10 words and 1 sentence, and the upper bound of the review was set to 7 sentences and 80 words. The goal here is to extract as much information from the reviews as possible. However, during the model training process, each sentence in a review is padded as the longest sentence in the review. Therefore, when the longest sentence in the review becomes too long, the padding will influence the performance of the model during training. In the experiments, the lower bound was set to (1, 20, 50) with the upper bound of (20, 50, 9999), this limit was only set for the sentence length in the training process, and the previous limitation [10 words, 3 sentences] and [80 words and 7 sentences] was the filter used in the selection of the review level. The selection strategy for the sentence length set as [20, 50] exhibited the best performance.
- The review corpus was document level based. In the postanalysis, however, each review in the corpus should be divided into sentences. In this study, Spacy [72] was utilized as the sentence divider.
- The review corpus comes with the same scoring system. Each user leaves a score from "1" to "5" when they giving the review. Since "1" and "2" are labeled as negative, "3" and "4" are labeled as neutral, and "5" is labeled as positive, the same number of reviews was selected for those three label categories to make the dataset more balanced. It is worth mentioning that in the raw dataset, two-star and one-star reviews have the smallest amounts. Therefore, to balance the sentiment labels, all two-star reviews (1185 reviews) and one-star reviews (2374 reviews) were selected to represent the "negative" sentiments.

There is a total of 145,430 reviews in the dataset, where 10,700 reviews have more than 60 words (long reviews), 22,458 reviews have less than 10 words (short reviews), and 1636 reviews mention product names. After removing all the long and short reviews to reduce the intrinsic bias, 59,184 reviews remained. Among the remaining reviews, 75.89% were five star, 12.85% were four star, 4.96% were three star, 2.64% were two star, and 4.92% were one star. In terms of attribute references, the "Exterior" and "Fit" attributes appeared the most in the raw dataset. Specifically, in all the reviews that contained attributes, 58.25% mentioned "Exterior," 76.88% mentioned "Fit," 12.21% mentioned "Shoe Parts," 32.11% mentioned "Durability," 7.33% mentioned "Permeability," 15.48% mentioned "Stability," and 16.59% mentioned "Impact absorption."

There are two models in the proposed methodology. This section presents the parameters used during the training process. In the MAS model, the parameters that can control the model performance includes batch size, learning rate, training steps, model dimension, and number of attention heads [83]. The definitions of these parameters are explained in the following with the T5 model. Several different parameters were tested during the training process. In the T5 fine-tuning process, the following parameters were adjusted to explore the best model performance:

- *Model*. The pretrained model utilized in the fine-tuning process.
- *Model dimension*. During the fine-tuning process, a dimension of 512 was used, which is the default dimension used in T5.
- *Keywords*. The MAS model can output synthetic summaries and keywords from the review. Those keywords could be used as input or output in the T5 model. In the experiment, the keywords were used as both inputs and outputs for the comparison.
- *Batch size*. The number of reviews trained in each iteration.
- *Learning rate*. The parameter that controls the parameter updating speed in the backpropagation process.
- *Training steps*. The parameter that controls the parameter updating time.
- *Learning rate scheduler*. The learning rate update controller that could change learning rate during training.
- *Minimum text length*. The minimum output length of the model.
- *Maximum text length*. The minimum output length of the model.

Table 1 The attribute lexicon [18]

Attribute category	Example attributes inside
Permeability	"ventilation," "breathable," "mesh"…
Impact absorption	"air," "gel," "strap",…
Stability	"flytrap," "ankle," "support",…
Durability	"durable," "ripple," "haptic",…
Shoe_Parts	"tonal," "bucket," "bottom",…
Exterior	"gold," "blocking," "metallic,"…
Fit	"dapper," "comfy," "adjustable,"…

As mentioned in the previous section, the early phase of the model utilized an attribute lexicon [18] to provide silver labels to the dataset. Part of the lexicon used in this research is presented in Table 1.

4.1 Model Performance.

In the model training process, several different sets of loss function and other hyper parameters were tested. The following is the loss function formula used in the experiment:

4.1.1 Soft-Margin Loss Function. This function creates a criterion that optimizes a two-class classification logistic loss between the input tensor y and the target tensor $\hat{y}$ (containing 1 or −1) [84]:

$$\mathcal{L}_{\mathrm{SM}} = \sum_j \frac{\log(1 + \exp(-\hat{y}_j y_j))}{y_n} \tag{11}$$

where j is the batch and n is the batch size.

4.1.2 Multilabel Soft-Margin Loss Function. This function creates a criterion that optimizes a multilabel one-versus-all loss based on max-entropy, between input y and target $\hat{y}$ of size (N, K) (K: number of classes). For each sample in the mini-batch:

$$\mathcal{L}_{\mathrm{MLSM}} = \frac{-1}{K}\sum_j \left(\hat{y}[j]\log(1 + \exp(-y[j]))^{-1} + (1 - \hat{y}[j])\log\frac{\exp(-y[j])}{1 + \exp(-y[j])}\right) \tag{12}$$

Table 2 Performance of the MAS model with different structures compared to the baseline model [61]

Metric	Learning rate	Loss function	Label type and activation function	Performance score (F1 in %)
MAS document—level F1	1e–5	Soft-margin (Eq. (11))	(1, −1) label with tanh	67.86
	1e–6	Soft-margin (Eq. (11))	(1, −1) label with tanh	70.91
	1e–6	Cross-entropy (Eq. (13))	(1, −1) label with tanh	74.53
	1e–6	Multilabel soft-margin (Eq. (12))	(1, −1) label with tanh	80.50
	1e–6	BCE mean weight (Eqs. (14) and (15))	(1, −1) label with tanh	79.41
	1e–6	Cross-entropy (Eq. (13))	(0, 1) label with ReLU	73.28
	1e–6	Multilabel soft-margin (Eq. (12))	(0, 1) label with ReLU	81.77
	1e–6	BCE mean weight (Eqs. (14) and (15))	(0, 1) label with ReLU	80.86
	1e–6	BCE sum weight (Eqs. (14) and (15))	(0, 1) label with ReLU	74.84
	1e–6	BCE calculated weight (Eqs. (14) and (16))	(0, 1) label with ReLU	**83.56**
MAS sentence—level F1	1e–5	Soft-margin (Eq. (11))	(1, −1) label with tanh	68.93
	1e–6	Soft-margin (Eq. (11))	(1, −1) label with tanh	70.93
	1e–6	Cross-entropy (Eq. (13))	(1, −1) label with tanh	74.44
	1e–6	Multilabel soft-margin (Eq. (12))	(1, −1) label with tanh	80.46
	1e–6	BCE mean weight (Eqs. (14) and (15))	(1, −1) label with tanh	78.27
	1e–6	Cross-entropy (Eq. (13))	(0, 1) label with ReLU	72.55
	1e–6	Multilabel soft-margin (Eq. (12))	(0, 1) label with ReLU	80.24
	1e–6	BCE mean weight (Eqs. (14) and (15))	(0, 1) label with ReLU	80.21
	1e–6	BCE sum weight (Eqs. (14) and (16))	(0, 1) label with ReLU	76.39
	1e–6	BCE calculated weight (Eqs. (14) and (16))	(0, 1) label with ReLU	**83.41**
Baseline document—level F1	1e–6	soft-margin (Eq. (11))	(−1, 1) label with tanh	74.65
	1e–6	BCE sum weight (Eqs. (14) and (16))	(0, 1) label with ReLU	76.54
Baseline sentence—level F1	1e–6	soft-margin (Eq. (11))	(1, −1) label with tanh	71.76
	1e–6	BCE sum weight (Eqs. (14) and (16))	(0, −1) label with tanh	73.74

Note: The bold text indicates the best performance in all the experiments.

4.1.3 Cross Entropy Loss Function. This function computes the cross-entropy loss between input and target. It is useful when training a classification problem with K classes, as follows:

$$\mathcal{L}_{\text{CE}} = -\sum_{k=1}^{K} w_k \log \frac{\exp(y_{n,k})}{\sum_{j=1}^{K} \exp(y_{n,j})} \hat{y_{n,k}} \tag{13}$$

where $\hat{y}$ is the target, y is the input, K is the label categories, k is one label in the label category, n is the batch size, and j is the sample in a batch.

4.1.4 Weighted Binary Cross-Entropy loss function. This function combines a Sigmoid layer and the BCE loss in one single class. This version is more numerically stable than using a plain Sigmoid followed by a BCE loss because combining the operations into one layer takes advantage of the log-sum-exp trick for numerical stability. The unreduced (i.e., with reduction set to "none") loss is as follows:

$$\mathcal{L}_{\text{BCE}} = -w_n[y_n \log \sigma(y_n) + (1 - \hat{y}_n) \log(1 - \sigma(y_n))] \tag{14}$$

where n is the batch size, $\hat{y}$ is the target, y is the input, and n is the batch size. If reduction is not "none" (default is "mean"), then:

$$\mathcal{L}_{\text{BCE}} = \begin{cases} \text{mean}(\mathcal{L}) & \text{if reduction } = \text{“mean”} \\ \text{sum}(\mathcal{L}) & \text{if reduction } = \text{“sum”} \end{cases} \tag{15}$$

Recall and precision can be balanced by adding weights to positive examples. In the case of multilabel classification, the loss can be formulated as follows:

$$\mathcal{L}_{n,k} = -w_{n,k}[p_k y_{n,k} \log \sigma(y_{n,k}) + (1 - y_{n,k}) \log(1 - \sigma(y_{n,k}))] \tag{16}$$

where $\hat{y}$ is the target, y is the input, n is the batch size, and k is the category. The weights assigned to the labels are 18.4872, 3.7189, 1.9146, 5.2818, 3.4055, 1.1269, 0.6156, 2.003, 1.985, and 2.009, which correspond to [Permeability, Impact Absorption, Stability, Durability, Shoe Parts, Exterior, Fit, Positive, Neutral, and Negative], respectively. The last three weights are the sentiment labels; since an equal number of reviews were selected for each sentiment label, the last three weights are almost equal.

The performance of the proposed MAS model is compared with a baseline model [61] in terms of the F1 score, as shown in Table 2. The last two rows compared the original model without sentiment information integrated with the original synthetic data creation method, using the same parameter setting. It is shown that our model outperforms the original model in terms of both sentence-level and document-level predictions. As observed in Table 2, the performance of both the MAS model and the baseline model varies with different loss functions. However, the proposed model is shown to outperform the baseline in terms of both document-level and sentence-level predictions, using the BCE loss and learning rate of 1e–6 for the ReLU activation function. This implies that the MAS model significantly improves both precision and recall in the classification of attributes and sentiments, compared to the baseline model [61].

In the T5 fine-tuning process, the ROUGE score was used as a benchmark to evaluate the performance of the model. The ROUGE-L score is based on the length of the longest common subsequence of candidates and references [81]. The ROUGE score performance of the sequence-to-sequence models is presented in Table 3. The T5 was fine-tuned with the same parameter setting in the baseline model and the MAS model. The best performance of the baseline T5 model (16.00) is slightly better than the proposed MAS-T5 model (15.67). This result was anticipated because, in the synthetic dataset creation process, the baseline model picks an entire user review as a synthetic summary, while the MAS-T5 model creates synthetic summaries by collecting useful sentences (i.e., sentences that contain attribute words) from different user reviews. Note that the ROUGE-L score measures the length of the longest common subsequence of candidates and references [81]. However, this compromise of the ROUGE-L score in the MAS-T5 model was necessary because many reviews contain contradictory sentiments, which in turn may "confuse" the summarization process if the whole review was used to compile the synthetic summaries, as performed in the baseline model [61].

Table 3 Performance of the sequence-to-sequence model with different structures compared to the baseline model [61]

Postmodel	Premodel	Learning rate	Training steps	Beam search size	ROUGE-L score
T5 small	Basline	1e−5	10,000	2	11.78
	MAS	1e−5	10,000	2	12.56
	MAS	1e−6	20,000	2	12.03
	MAS	1e−6	50,000	2	15.21
	MAS	1e−6	100,000	2	**15.54**
	MAS	1e−6	10,000	3	11.33
	MAS	1e−6	20,000	3	12.25
	MAS	1e−6	50,000	3	15.06
	MAS	1e−6	100,000	3	15.11
	MAS	1e−6	100,000	4	15.24
T5 s v_1_1	Basline	1e−5	10,000	2	12.11
	MAS	1e−6	10,000	2	11.34
	MAS	1e−6	20,000	2	12.57
	MAS	1e−6	50,000	2	**15.67**
	MAS	1e−6	100,000	2	15.63
	MAS	1e−6	10,000	3	11.27
	MAS	1e−6	20,000	3	12.56
	MAS	1e−6	50,000	3	15.23
	MAS	1e−6	100,000	3	15.34
	MAS	1e−6	100,000	4	15.66
T5 small	Basline	1e−5	10,000	2	11.78
	Basline	1e−6	50,000	2	16.00

Note: The bold text indicates the best performance in all the experiments.

4.2 Opinion Polarity and Subjectivity. The opinion polarity and subjectivity score patterns are analyzed for each attribute. From the collection of all reviews. For each attribute, a similarity score was determined for each review with reference to the set of representative keywords for that attribute. The set of ten reviews with the highest similarity scores was selected for that attribute. The opinion polarity and subjectivity were determined for that subset in this case, and the subset means the review filter mentioned in Sec. 3. The distribution of the opinion polarity scores for each attribute is presented in Fig. 3 as box plots. Polarity scores seemed to be in a similar range for most attributes with mean vaswitch = [1,1,1,1,1,1,1,1,0,0]lues around 0.2, indicating an overall positive nature of the reviews. In Fig. 5, a relatively higher number of negative polarity data points can be observed for "exterior" and "stability" attributes compared to other attributes. The attribute-wise distribution of the subjectivity scores of the opinion is presented as box plots in Fig. 6. The subjectivity score distribution also appeared to be similar for most attributes, with a mean value of around 0.5, indicating that most of the reviews were reasonably objective in nature.

The distribution of opinion polarity and subjectivity is also compared in the original collection of reviews and in the collection of review summaries to check if there were any deviations. For each of the reviews collected for each attribute, the summary was generated using the T5 model. The opinion polarity and subjectivity scores were calculated for each review in the original collection and the summary generated. The comparative analysis between the distribution of these scores in the collection and the summary provides information on any loss of information. Figure 7 shows the distribution of opinion polarity and subjectivity for the original review set and summary set. It can be observed that the variance in polarity and subjectivity distributions was relatively less for the summary set compared to original reviews, which seems intuitive as the original review dataset was more noisy. We can also observe in Fig. 7 that the mean values of opinion polarity and subjectivity scores were relatively higher for the summary dataset compared to the original review dataset. For polarity, the mean of the summary data set was 0.32, while the mean of the review data set was 0.18. For subjectivity, the mean of the summary dataset was 0.58, while the review dataset mean was 0.52. Based on the unequal variance t-test, the differences in mean polarity and subjectivity scores between the summary and review datasets were found to be statistically significant. This difference in polarity and subjectivity can be qualitatively further analyzed to understand the nature of information loss in the summarization process.

4.3 Attribute-Sentiment-Guided Summaries. In the actual model training process, the word "comfortable" has the highest possibility in the "positive" summarization. To avoid imbalanced results, "comfortable" was removed from the attribute lexicon during training. Some attribute-sentiment summarization results are shown below. Example of "positive" and "color" summarization

> "i have always loved air force 1s so this is so much better. i love the look with jeans, the icy blue is more of a plain gray, and more important the laces are white - not a light blue as shown. meanwhile i bought these for my son because he wanted this style and he was having an all white dance."

Examples of "positive" and "durability" summarization

> "i love these shoes. i'm a healthcare worker and i have been wearing them for a long time and they are very comfortable and comfortable. they are a great fit and a good fit! i bought these for my son and he loves them."
>
> "i'm a nurse and i've been a fan of these shoes. i have been wearing these shoes for a while now. they're a great shoe for a long time."
>
> "i'm a walker and i've been able to wear them all day. i love them! i love these shoes! they are so comfortable and comfortable. i have been wearing them for a while now. i can't wait to see if they'll be available."

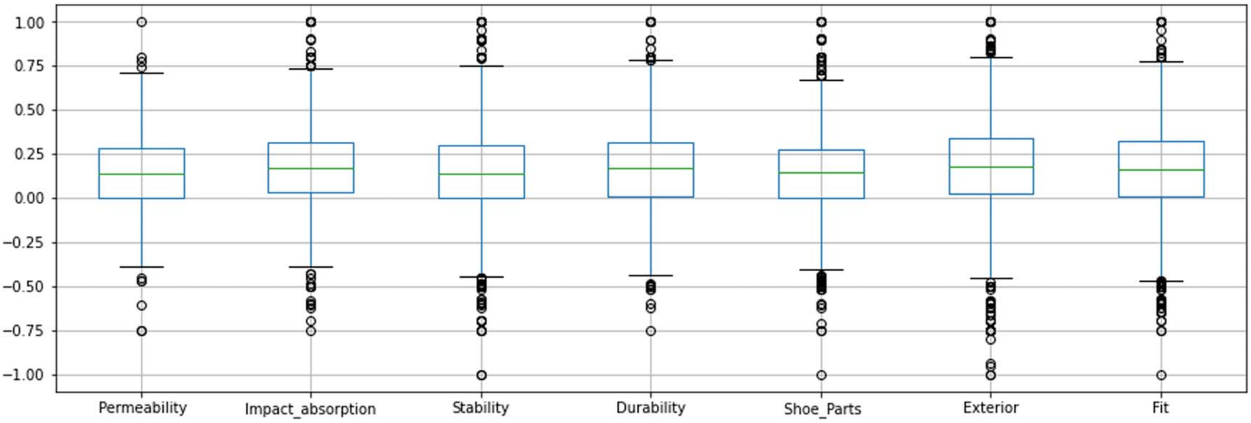

Fig. 5 Polarity score distribution for different attributes

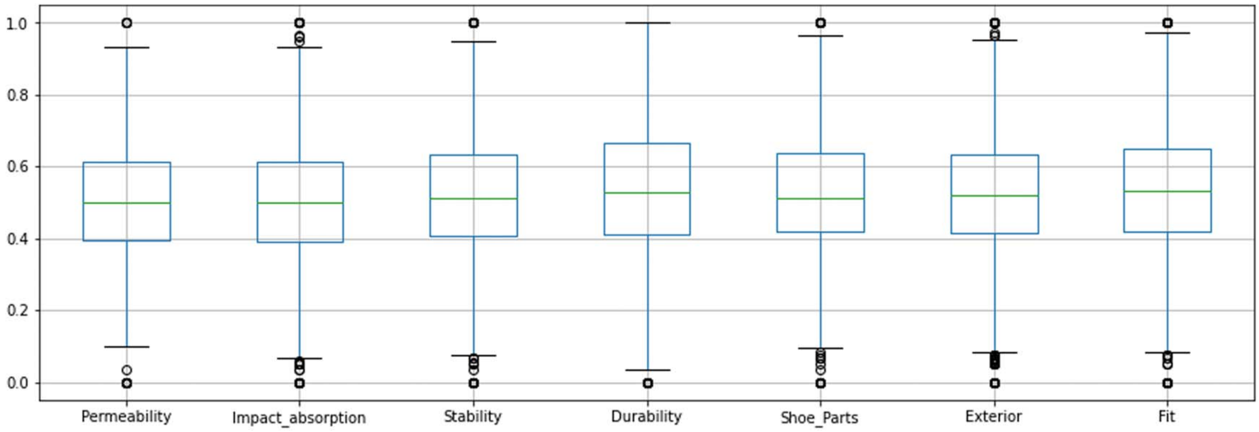

Fig. 6 Subjectivity score distribution for different attributes

Examples of "positive" and "fit" summarization

> "a great shoe! i have a a sleeve and they are a great fit! i love the color of the shoe. i love how they feel on my feet. they are so comfortable."
> "i'm wearing these shoes for a long time. i love the fit. they are very comfortable."

Examples of "negative" and "shoe parts" summarization

> "the toe box was uncomfortably huge. i have always loved the reebok classics. hurt the the side of my foot."
> "they are too wide! i have to find another pair, i have a couple of pairs and the shoe sleeve is easy to move."

Examples of "negative" and "color" summarization

> "the color of the grey is not what I expected. i have always love this sneaker, the color is not what i saw in the picture, the color is darker on the top."
> "the silver strip on the side is boring, the blue heel is not the right blue, the heel is tough."

Some interesting observations can be immediately made from the aforementioned example results. For example, when people review a product, they often refer to its usage context as well. When the users of a footwear item say something about its durability, for example, they also mention their occupations, which may require standing for long periods of time (e.g., nurses, doctors, factory workers). Moreover, it can be observed that negative reviews usually tend to address specific parts of the item (e.g., a small or tight toe box, hard heels) or a specific attribute not matching the original description on the website (e.g., online versus actual color). Since the generated summaries capture the most informative parts of the reviews, designers can confidently rely on the summaries generated with respect to different sentiments and/or attributes to quickly evaluate any potential relationships between the causes of dissatisfaction and compare different competing items on the market when designing a new concept.

The presented model only utilizes the ROUGE score for training. However, this benchmark only pays attention to the overlapping rate of the model output and the human-generated summary. To potentially generate more useful information for designers, the ROUGE score may not be sufficient since in the review content, the most common part is probably not expressed in the *design language*. For example, the most common sentences in the positive review summaries generated in this work are "this sneaker is very comfortable" and "the shoe is very comfortable". Hence, during the fine-tuning process, the model will learn to include sentences

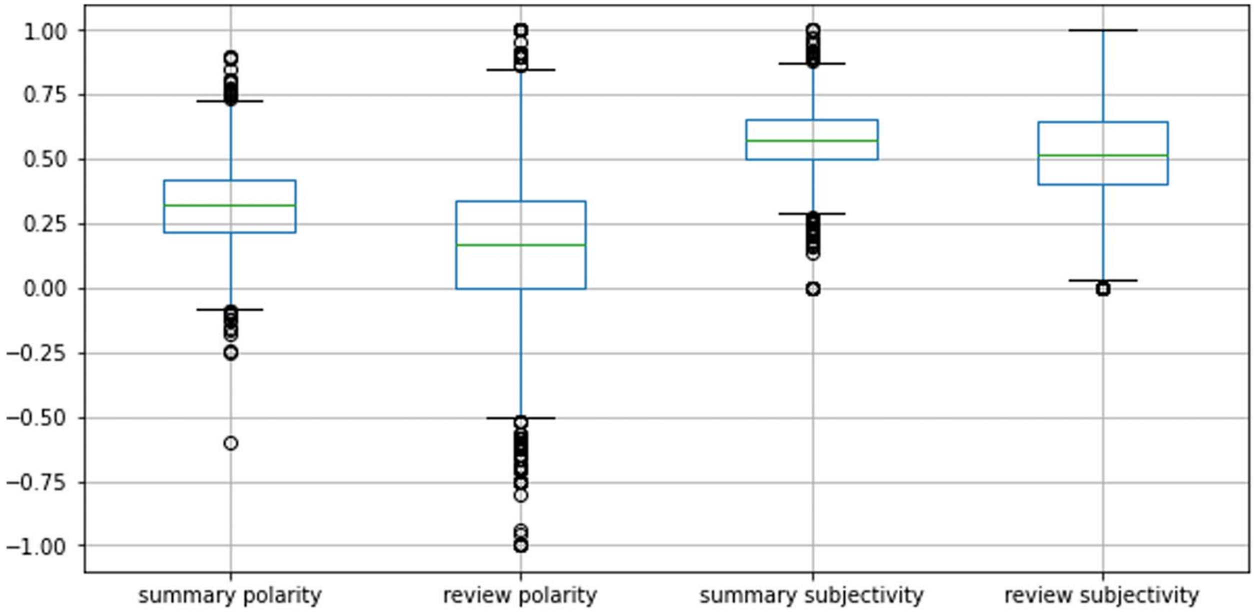

Fig. 7 Polarity and subjectivity score distributions in reviews and summary datasets

like these in the summary with a higher probability. Yet, this kind of summary may have very limited implications for designers. Another limitation related to the design language comes from the review corpus itself. Specifically, users barely provide professional design feedback in their positive reviews, but tend to be very specific and helpful in their negative reviews. This fact makes summarization from positive reviews usually not capable of generating useful outputs and recommendations for designers.

5 Conclusions and Future Research Directions

This article proposed a novel MAS-T5 framework for the automated and large-scale generation of opinion summaries from online reviews, guided by user sentiments and product attributes. Building on advanced NLP research on language models, the framework is anticipated to save significant amounts of time and effort for data preparation and reduce the need for hand-engineered expert systems for opinion summarization from reviews. The developed framework also enables an efficient and continuous update of the opinion extraction results as more users publish their feedback and opinions on e-Commerce platforms on a daily basis. The advantages of the MAS-T5 framework for the large-scale attribute-sentiment-guided opinion summarization are as follows:

- *Efficiency and scalability*. The use of pretrained language models such as T5 reduces the need for large manually labeled data. All components of the MAS-T5 methodology are packaged in structured fashions and can be quickly modified and applied to any sentiment-oriented opinion summarization problem. The methodology also reduces the need to conduct extensive market studies, focus groups, interviews, or lead user analyses. The only input required is free text, user reviews. Consequently, the investments required in human power, budget, and space to conduct large-scale need finding studies can be dramatically reduced.
- *Automated and large-scale sentiment-oriented opinion summary*. The MAS-T5 methodology extracts an exhaustive list of candidate attribute sentiment-oriented summaries. The summarization process developed provides strong flexibility to switch target sentiment and attribute. This is a significant step toward enabling automated and large-scale opinion summarization, which, compared to lead user-based approaches, can potentially extract more informative and potentially transformative insights to inform the design process.
- *Modular structure*. The MAS-T5 network is composed of independent modules for the analysis of the MAS and T5. Such a modular structure enables flexibility for independent/parallel improvement of different modules with new add-on features for handling different NLP tasks.

5.1 Implications for Product Design.

Opinion summarization has not been widely explored as a means to inform the design of new products. The elicitation and incorporation of user data in the design process have been shown to be effective for the overall success of new product/service development processes [2] by increasing the quantity and quality of ideas at the front-end of the design process [3,4]. There is a substantial opportunity to improve the front-end of design innovation processes by generating brief, guided summaries of user feedback from myriad reviews available on various e-Commerce and social media platforms. This article builds on the state-of-the-art in deep language representation [8,9] and information extraction [85–87] to generate selective and filtered summaries of attribute-specific user sentiments that currently cannot be manually processed by designers due to the large quantity and diversity of reviews. The existing research that attempted to bridge this gap uses most of the information extraction to select reviews with the goal of filtering useful reviews for designers [88,89]. However, thousands of reviews remain even after the review selection process, and the question of how to summarize them into shorter, more guided executive summaries becomes more and more important. Some researchers have also tried to identify useful keywords from the review corpus, but these methods still lack detailed design information [90,91]. All of these limitations and potentials point to the importance of guided and controllable opinion summarization for early-stage product development. To fully realize the potentials of the proposed methodology, extensive future research is required, on both methodology and validation, to extract complex, nonobvious, and difficult-to-identify user opinions and ideally, the *latent needs*.

5.2 Future Research: Methodology.

Further research should optimize the MAS-T5 network architecture for reducing loss and improving rouge score. The main technical limitations of the MAS-T5 methodology to be improved in future work are summarized as follows:

- *Low-level ROUGE score*. The free-text reviews used as input in this work are noisy and imbalanced. Moreover, the training dataset used to fine-tune the T5 model only included synthetic summaries from the original corpus. Therefore, the training dataset inherently contained some incoherence in the expression. The baseline model [61] has a similar best performance to the presented model in terms of the ROUGE-L score. The synthetic data creation method was the main disturbance factor in the ROUGE-L score. In our case, the sentiment label and the attribute label are used as the indicator of sentence selection in the synthetic data creation process, breaking the entire review into sentences to assemble the synthetic summary will instinctively reduce performance in terms of the ROUGE score.
- *Lack of attributes and sentiment*. The raw dataset was highly imbalanced with respect to both attributes and sentiments. Specifically, among all the reviews in the corpus, over 92% users gave five star ratings, even though some of them complained about the product. In terms of attributes, fit, shoe parts, and exterior together represented more than 90% of the attributes mentioned in the original corpus. Further, in the word section of the model prediction, the word “comfortable” appeared ten times more frequently than the second most frequent word.
- *Potential information loss due to long network structure*. The MAS-T5 network comprises several submodels for MAS, synthetic data creation, and T5 fine-tuning. Each submodel has a separate loss, which in turn may cause information loss and make the output very noisy. This problem can be addressed by improving the architecture of each submodel.
- *The lack of human-annotated dataset*. In the baseline model [61], authors use a human-annotated dataset to test and validate their model. However, the presented work did not use any human-annotated dataset. The model can therefore be improved in the future through testing and validation using a human-annotated dataset.

5.3 Future Research: Validation.

It is not yet clear whether and how the proposed methodology will impact the performance of the design team in finding meaningful and informative user opinion summaries in practice. Future research must conduct extensive studies on humans in controlled laboratory environments to measure the difference assumed between the performance of a design team using the MAS-T5 results and another design team reading reviews directly from e-commerce platforms. Future research must devise new mechanisms to measure how informative each identified summary is to the designer. That is, even if the results are guided with respect to attributes and sentiments, there are still various other ways to organize and analyze opinion summaries. Finally, professional designers must be involved in the process of building and validating these models to ensure effective practical use.

Acknowledgment

This material is based on the work supported by the National Science Foundation under the Engineering Design and System Engineering (EDSE) (Grant No. #2050052). Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Conflict of Interest

There are no conflicts of interest.

Data Availability Statement

The datasets generated and supporting the findings of this article are obtainable from the corresponding author upon reasonable request.

Nomenclature

a = element wise product of each key and query
b = constant added to the linear and nonlinear transformation of the word encoding
c = label categories in the MAS model
e = encoding from pretrained model
h = attention head in the MAS model
k = kth head in the MAS model
n = batch size in the MAS model
p = probability sample drawn from a Bernoulli distribution
y = actual label
A = attribute word set
C = review corpus
$\mathcal{S}$ = similarity between each sentence and the summary
$\hat{y}$ = predicted label
a_n = nth attribute word in A
a_h = attention output in the MAS model
b_h = constant added to the linear and nonlinear transformation of the attention head encoding
r_i = ith review in a review batch
$w_{n,c}$ = weights of category c
$\mathcal{L}_{\text{MAS}}$ = loss function of the MAS model
$\mathcal{L}_{\text{sum}}$ = loss function of the sequence to sequence model
P_a = attribute label prediction in the MAS model
$\mathcal{P}_c$ = category's weight in the MAS model
P_h = head prediction in the MAS model
P_r = review level prediction in the MAS model
P_s = sentence-level prediction in the MAS model
P_{sa} = sentence-level attribute prediction in the MAS model
P_{ss} = sentence-level sentiment prediction in the MAS model
P_t = token-level prediction in the MAS model
W_i = ith word in a sentence
W_e = word encoding result after using the RoBERTa encoder
W_{he} = attention head encoding result after using the RoBERTa encoder
W_n = word list of a review, including n words

Appendix

Table 4 The attribute lexicon [18]

Category	Attributes
Permeability	“permeability,” “ventilation,” “breathable,” “mesh,” “nylon,” “zoned,” “forged,” “perforated,” “chamois,” “adaptive,” “neoprene,” “pigskin,” “rubber,” “waterproof,” “construction,” “coating,” “pod,” “repellent,” “leather,” “insulation,” “rustproof,” “forefoot,” “resistant,” “textile,” “lining,” “membrane,” “breathable”
Impact absorption	“impact absorption,” “supportive,” “air,” “gel,” “strap,” “foam,” “bounce,” “shock,” “segmented,” “geometric,” “pattern,” “zoom,” “energy,” “compression,” “flex,” “impact,” “guidance,” “react,” “protection,” “loft,” “vertical,” “groove,” “energy return,” “flair,” “propulsion,” “reflective,” “boost,” “turbo,” “embroidery”
Stability	“stability,” “warmth,” “grip,” “heel,” “clip,” “lateral,” “synthetic,” “continental,” “collar,” “underlay,” “cage,” “barrier,” “fusible,” “knit,” “fabric,” “sticky,” “torsion,” “bungee,” “tape,” “smooth,” “ride,” “wedge,” “external,” “flytrap,” “ankle,” “support,” “carbon,” “fiber,” “guide,” “tongue,” “flexibility,” “flexible,” “stretchy,” “gore,” “panel,” “phylon,” “speedy,” “explosive,” “graphic,” “wear,” “traction,” “abrasion,” “solid,” “herringbone,” “waffle,” “circular,” “multidirectional,” “rugged,” “tread,” “canvas,” “knobbed,” “chevron,” “sponge,” “lug”
Durability	“durability,” “drier,” “breezy,” “cooler,” “suede,” “tumbled,” “lightweight,” “vamp,” “durable,” “ripple,” “haptic,” “thin,” “woven,” “material,” “overlay”
Shoe parts	“cushy,” “fusion,” “firm,” “absorbing,” “springy,” “poly,” “wavy,” “padding,” “speckled,” “translucent,” “cut,” “tonal,” “grippy,” “bottom,” “bold,” “curvy,” “removable,” “cushiony,” “thick,” “hard,” “soft,” “exoskeletal,” “beveled,” “iridescent,” “silhouette,” “low,” “sheen,” “skin,” “covert,” “exoskeletal,” “bucket,” “lacing,” “zone,” “saddle,” “cushion,” “elastic,” “cushioned,” “optimal,” “plush,” “cotton,” “responsive,” “insole,” “ignite,” “visible,” “pillowy,” “fixation,” “sassy,” “toggle,” “loop,” “laceless,” “zip,” “gilly,” “asymmetrical,” “magnetic,” “buckle,” “iconic,” “lace,” “futuristic,” “cap,” “tuff,” “embellishment,” “clasp,” “apparel,” “welt,” “quilted,” “posture,” “eyelet,” “solar”
Exterior	“color,” “red,” “yellow,” “blue,” “striking,” “graphics,” “palette,” “gold,” “blocking,” “metallic,” “marble,” “black,” “orange,” “anthracite,” “white,” “royal,” “gloss,” “stripe,” “sweeping,” “shape,” “wardrobe,” “arch,” “sleek,” “structural,” “flattering,” “edgy,” “masculine,” “anatomic,” “nib,” “versatile,” “exaggerated,” “inflated,” “swoosh,” “chunky,” “bulky,” “style,” “boat,” “tall,” “doodle,” “look,” “zipper,” “stitching,” “shearling,” “calf,” “strapless,” “insulated,” “patchwork,” “foxing,” “washable,” “topline,” “surface,” “stretch,” “ribbing,” “asymmetric,” “yarn,” “plastic,” “stretchable,” “melange,” “exposed,” “paneling”
Fit	“trim,” “gait,” “big,” “small,” “dress,” “distressed,” “dapper,” “comfy,” “adjustable,” “narrow,” “custom,” “strategic,” “large,” “closure,” “curved,” “inner,” “sleeve,” “secure,” “snug,” “comfortable,” “band,” “crisscross,” “wide,” “width,” “softfoam,” “anatomical,” “holistic,” “weight,” “heavy,” “light,” “featherweight”

References

[1] Schaffhausen, C. R., and Kowalewski, T. M., 2015, "Large-Scale Needfinding: Methods of Increasing User-Generated Needs From Large Populations," ASME J. Mech. Des., **137**(7), p. 071403.

[2] Cooper, R. G., Edgett, S. J., and Kleinschmidt, E. J., 2004, "Benchmarking Best NPD Practices—III," Res. Tech. Manage., **47**(6), pp. 43–55.

[3] Osborn, A. F., 1953, *Applied Imagination*, Scribner's, New York.

[4] Marion, T. J., and Fixson, S. K., 2018, *The Innovation Navigator: Transforming Your Organization in the Era of Digital Design and Collaborative Culture*, University of Toronto Press, Canada.

[5] Eckert, C., 1999, "Managing Effective Communication in Knitwear Design," Des. J., **2**(3), pp. 29–42.

[6] Rasoulifar, G., Eckert, C., and Prudhomme, G., 2015, "Communicating Consumer Needs in the Design Process of Branded Products," ASME J. Mech. Des., **137**(7), p. 071404.

[7] Franke, N., Schreier, M., and Kaiser, U., 2010, "The 'I Designed It Myself' Effect in Mass Customization," Manage. Sci., **56**(1), pp. 125–140.

[8] Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I., 2018, "Improving Language Understanding by Generative Pre-Training," Technical Report.

[9] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K., 2018, "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint.

[10] Mirtalaie, M. A., Hussain, O. K., Chang, E., and Hussain, F. K., 2017, "A Decision Support Framework for Identifying Novel Ideas in New Product Development From Cross-Domain Analysis," Inform. Syst., **69**(C), pp. 59–80.

[11] Pang, B., and Lee, L., 2006, "Opinion Mining and Sentiment Analysis.," Found. Trends Information Retrieval, **1**(2), pp. 91–231.

[12] Tang, H., Tan, S., and Cheng, X., 2009, "A Survey on Sentiment Detection of Reviews," Expert. Syst. Appl., **36**(7), pp. 10760–10773.

[13] Zhang, L., Wang, S., and Liu, B., 2018, "Deep Learning for Sentiment Analysis: A Survey," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, **8**(4), p. e1253

[14] Liu, B., 2020, *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*, Vol. 38, The Cambridge University Press, Cambridge, MA, pp. 41–51.

[15] Ireland, R., and Liu, A., 2018, "Application of Data Analytics for Product Design: Sentiment Analysis of Online Product Reviews," CIRP. J. Manuf. Sci. Technol., **23**, pp. 128–144.

[16] Nasukawa, T., and Yi, J., OCT 2003, "Sentiment Analysis: Capturing Favorability Using Natural Language Processing," Proceedings of the 2nd International Conference on Knowledge Capture, pp. 70–77.

[17] Han, Y., and Moghaddam, M., 2020, "Eliciting Attribute-Level User Needs From Online Reviews With Deep Language Models and Information Extraction," ASME J. Mech. Des., **143**(6), p. 061403.

[18] Han, Y., and Moghaddam, M., 2021, "Analysis of Sentiment Expressions for User-Centered Design," Expert Syst. Appl., **171**, p. 114604.

[19] Zheng, H., and Lapata, M., 2019, "Sentence Centrality Revisited for Unsupervised Summarization," arXiv preprint.

[20] Cachola, I., Lo, K., Cohan, A., and Weld, D. S., 2020, "TLDR: Extreme Summarization of Scientific Documents," arXiv preprint.

[21] Angelidis, S., and Lapata, M., 2018, "Summarizing Opinions: Aspect Extraction Meets Sentiment Prediction and They Are Both Weakly Supervised," arXiv preprint.

[22] Ganesan, K., Zhai, C. X., and Han, J., 2010, "Opinosis: A Graph Based Approach to Abstractive Summarization of Highly Redundant Opinions," ACL Anthology, pp. 340–348.

[23] Di Fabbrizio, G., Stent, A., and Gaizauskas, R., 2014, "A Hybrid Approach to Multi-Document Summarization of Opinions in Reviews," ACL Anthology, pp. 54–63.

[24] Liu, Y., and Lapata, M., 2019, "Text Summarization With Pretrained Encoders," ACL Anthology, pp. 3730–3740.

[25] Ray, P., and Chakrabarti, A., 2020, "A Mixed Approach of Deep Learning Method and Rule-Based Method to Improve Aspect Level Sentiment Analysis," Appl. Comput. Inform.

[26] Timoshenko, A., and Hauser, J. R., 2019, "Identifying Customer Needs From User-Generated Content," Market. Sci, **38**(1), pp. 1–20.

[27] Bohm, M. R., and Stone, R. B., 2008, "Product Design Support: Exploring a Design Repository System," ASME International Mechanical Engineering Congress and Exposition, American Society of Mechanical Engineers Digital Collection, Vol. 47047, pp. 55–65.

[28] Lu, Y. Q., Liu, P.-L., Ding, X.-M., and Fu, Q.-R., 2008, "Plastic Product Evaluation Based on Mold Conceptual Design," ASME International Mechanical Engineering Congress and Exposition, American Society of Mechanical Engineers Digital Collection, Vol. 42150, pp. 319–327.

[29] Chen, X., Sun, C., Wang, J., Li, S., Si, L., Zhang, M., and Zhou, G., 2020, "Aspect Sentiment Classification With Document-Level Sentiment Preference Modeling," Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 3667–3677.

[30] Wang, K., Shen, W., Yang, Y., Quan, X., and Wang, R., 2020, "Relational Graph Attention Network for Aspect-Based Sentiment Analysis," arXiv.

[31] Rietzler, A., Stabinger, S., Opitz, P., and Engl, S., 2019, "Adapt or Get Left Behind: Domain Adaptation Through BERT Language Model Finetuning for Aspect-Target Sentiment Classification," arXiv.

[32] Yu, J., and Jiang, J., 2019, "Adapting BERT for Target-Oriented Multimodal Sentiment Classification," IJCAI, p. 5408.

[33] Karimi, A., Rossi, L., and Prati, A., 2020, "Adversarial Training for Aspect-Based Sentiment Analysis with BERT," 25th International Conference on Pattern Recognition (ICPR), IEEE, pp. 8797–8803.

[34] Hoang, M., Bihorac, O. A., and Rouces, J., 2019, "Aspect-Based Sentiment Analysis Using BERT," Proceedings of the 22nd Nordic Conference on Computational Linguistics, pp. 187–196.

[35] Xu, H., Liu, B., Shu, L., and Yu, P. S., 2019, "BERT Post-Training for Review Reading Comprehension and Aspect-Based Sentiment Analysis," arXiv preprint .

[36] Ma, Y., Peng, H., and Cambria, E., Apr 2018, "Targeted Aspect-Based Sentiment Analysis via Embedding Commonsense Knowledge Into an Attentive LSTM," Proceedings of the AAAI conference on artificial intelligence, Vol. 32, No. 1.

[37] Dai, Z., and Huang, R., 2021, "A Joint Model for Structure-based News Genre Classification with Application to Text Summarization," Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 3332–3342.

[38] Sun, E., Hou, Y., Wang, D., Zhang, Y., and Wang, N. X. R., 2021, "D2S: Document-to-Slide Generation Via Query-Based Text Summarization," arXiv.

[39] Liu, Y., Shen, S., and Lapata, M., 2020, "Noisy Self-Knowledge Distillation for Text Summarization," arXiv.

[40] Liu, Y., and Lapata, M., 2019, "Text Summarization With Pretrained Encoders," arXiv.

[41] Shao, Z., Bian, H., Chen, Y., Wang, Y., Zhang, J., Ji, X., and Zhang, Y., 2021, "TransMIL: Transformer Based Correlated Multiple Instance Learning for Whole Slide Image Classification," Adv. Neural Inf. Process Syst., **34**, pp. 2136–2147.

[42] Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Jozefowicz, R., and Bengio, S., 2015, "Generating Sentences From a Continuous Space," arXiv.

[43] See, A., Liu, P. J., and Manning, C. D., 2017, "Get to the Point: Summarization With Pointer-Generator Networks," arXiv.

[44] Bražinskas, A., Lapata, M., and Titov, I., 2019, "Unsupervised Opinion Summarization as Copycat-Review Generation," arXiv.

[45] Lin, C.-Y., 2004, "Rouge: A Package for Automatic Evaluation of Summaries," Text Summarization Branches Out, pp. 74–81.

[46] Cavnar, W. B., and Trenkle, J. M., 1994, "N-gram-Based Text Categorization," Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval (Vol. 161175).

[47] Nayeem, M. T., Fuad, T. A., and Chali, Y., 2018, "Abstractive Unsupervised Multi-Document Summarization Using Paraphrastic Sentence Fusion," Proceedings of the 27th International Conference on Computational Linguistics, pp. 1191–1204.

[48] Qiu, Y., and Jin, Y., 2021, "Engineering Document Summarization Using Sentence Representations Generated by Bidirectional Language Model," International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 85376, American Society of Mechanical Engineers, p. V002T02A062.

[49] Liu, Y., and Lapata, M., 2019, "Hierarchical Transformers for Multi-Document Summarization," arXiv preprint, pp. 5070–5081.

[50] Nayeem, M. T., Fuad, T. A., and Chali, Y., 2018, "Abstractive Unsupervised Multi-Document Summarization Using Paraphrastic Sentence Fusion," Proceedings of the 27th International Conference on Computational Linguistics, pp. 1191–1204.

[51] Jin, H., Wang, T., and Wan, X., 2020, "Multi-Granularity Interaction Network for Extractive and Abstractive Multi-Document Summarization," Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 6244–6254.

[52] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J., 2019, "Exploring the Limits of Transfer Learning With a Unified Text-to-Text Transformer," J. Mach. Learn. Res., **21**(140), pp. 1–67.

[53] Schumann, R., Mou, L., Lu, Y., Vechtomova, O., and Markert, K., 2020, "Discrete Optimization for Unsupervised Sentence Summarization with Word-Level Extraction," arXiv.

[54] Pugoy, R. A., and Kao, H.-Y., 2021, "Unsupervised Extractive Summarization-Based Representations for Accurate and Explainable Collaborative Filtering," Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 2981–2990.

[55] Kingma, D. P., and Ba, J., 2014, "Adam: A Method for Stochastic Optimization," arXiv.

[56] Angelidis, S., and Lapata, M., 2018, "Summarizing Opinions: Aspect Extraction Meets Sentiment Prediction and They Are Both Weakly Supervised," arXiv.

[57] Angelidis, S., Kim Amplayo, R., Suhara, Y., Wang, X., and Lapata, M., 2021, "Extractive Opinion Summarization in Quantized Transformer Spaces," Trans. Assoc. Comput. Ling., **9**, pp. 277–293.

[58] Chu, E., and Liu, P., 2019, "MeanSum: A Neural Model for Unsupervised Multi-Document Abstractive Summarization," International Conference on Machine Learning, PMLR, pp. 1223–1232.

[59] Amplayo, R. K., Angelidis, S., and Lapata, M., 2021, "Unsupervised Opinion Summarization With Noising and Denoising," Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 14, pp. 1934–1945.

[60] Elsahar, H., Coavoux, M., Rozen, J., and Gallé, M., 2021, "Self-Supervised and Controlled Multi-Document Opinion Summarization," arXiv preprint, pp. 1646–1662.

[61] Amplayo, R. K., Angelidis, S., and Lapata, M., 2021, "Aspect-Controllable Opinion Summarization," arXiv.

[62] Mani, I., Maybury, M. T., Maybury, M. T., and Maybury, M., 1999, *Advances in Automatic Text Summarization*, MIT Press, MA.

[63] Hernández-Castañeda, Á., García-Hernández, R. A., Ledeneva, Y., and Millán-Hernández, C. E., 2020, "Extractive Automatic Text Summarization Based on Lexical-Semantic Keywords," IEEE Access, **8**, pp. 49896–49907.

[64] Li, C., Xu, W., Li, S., and Gao, S., 2018, "Guiding Generation for Abstractive Text Summarization Based on Key Information Guide Network," Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pp. 55–60.

[65] Lee, T. Y., and Bradlow, E. T., 2011, "Automated Marketing Research Using Online Customer Reviews," J. Market. Res., **48**(5), pp. 881–894.

[66] Ravi, K., and Ravi, V., 2015, "A Survey on Opinion Mining and Sentiment Analysis: Tasks, Approaches and Applications," Knowledge Based Syst., **89**, pp. 14–46.

[67] Yadav, N., and Chatterjee, N., 2016, "Text Summarization Using Sentiment Analysis for DUC Data," 2016 International Conference on Information Technology (ICIT), IEEE, pp. 229–234.

[68] Musto, C., Rossiello, G., de Gemmis, M., Lops, P., and Semeraro, G., 2019, "Combining Text Summarization and Aspect-Based Sentiment Analysis of Users' Reviews to Justify Recommendations," Proceedings of the 13th ACM Conference on Recommender Systems, pp. 383–387.

[69] Mirani, T. B., and Sasi, S., 2017, "Two-Level Text Summarization From Online News Sources With Sentiment Analysis," 2017 International Conference on Networks & Advances in Computational Technologies (NetACT), IEEE, pp. 19–24.

[70] Alsaqer, A. F., and Sasi, S., 2017, "Movie Review Summarization and Sentiment Analysis Using Rapidminer," 2017 International Conference on Networks & Advances in Computational Technologies (NetACT), IEEE, pp. 329–335.

[71] Tsai, C.-F., Chen, K., Hu, Y.-H., and Chen, W.-K., 2020, "Improving Text Summarization of Online Hotel Reviews With Review Helpfulness and Sentiment," Tourism Manage., **80**, p. 104122.

[72] "spaCy, Industrial-Strength Natural Language Processing in Python," https://spacy.io, Accessed January 23, 2022.

[73] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K., 2018, "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding," arXiv preprint.

[74] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I., 2017, "Attention Is All You Need," Adv. Neural Inf. Process Syst., **30**.

[75] NLTK, 2022, "Natural Language Toolkit, https://www.nltk.org, Accessed July 31, 2022.

[76] "spaCy, Industrial-Strength Natural Language Processing in Python, https://spacy.io, Accessed July 31, 2022.

[77] Ramesh, G. S., Manyam, V., Mandula, V., Myana, P., Macha, S., and Reddy, S., 2022, "Abstractive Text Summarization Using T5 Architecture," Proceedings of Second International Conference on Advances in Computer Engineering and Communication Systems, Springer, Singapore, pp. 535–543.

[78] Agrawal, Y., Thakre, A., Tapas, T., Kedia, A., Telkhade, Y., and Rathod, V., 2021, "Comparative Analysis of NLP Models for Google Meet Transcript Summarization," EasyChair Preprint, (5404).

[79] Bohra, M., Dadure, P., and Pakray, P., 2022, "Comparative Analysis of T5 Model for Abstractive Text Summarization on Different Datasets."

[80] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R., 2012, "Improving Neural Networks by Preventing Co-Adaptation of Feature Detectors," arXiv.

[81] Lin, C.-Y., 2014, "ROUGE: A Package for Automatic Evaluation of Summaries," Text Summarization Branches Out, pp. 74–81.

[82] TextBlob, 2021, "Simplified Text Processing—TextBlob 0.16.0 Documentation," https://textblob.readthedocs.io/en/dev, Accessed August 3, 2022.

[83] Vaswani, A., Shazeer,, N., Parmer, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I., 2017, "Attention is all you Need," Adv. Neural Inf. Process Syst., **30**.

[84] Shawe-Taylor, J., and Cristianini, N., 2002, "On the Generalization of Soft Margin Algorithms," IEEE Trans. Inf. Theory, **48**(10), pp. 2721–2735.

[85] Nadeau, D., and Sekine, S., 2007, "A Survey of Named Entity Recognition and Classification," Lingvisticæ Investigationes, **30**(1), pp. 1–20.

[86] Li, X., Bing, L., Li, P., Lam, W., and Yang, Z.,, 2018, "Aspect Term ExtractionWith History Attention and Selective Transformation," arXiv preprint.

[87] Yadav, V., and Bethard, S., 2019, "A Survey on Recent Advances in Named Entity Recognition From Deep Learning Models," arXiv.

[88] Liu, Y., Jin, J., Ji, P., Harding, J. A., and Fung, R. Y. K., 2013, "Identifying Helpful Online Reviews: A Product Designer's Perspective," Comput. Aided Des., **45**(2), pp. 180–194.

[89] Li, X., and Hitt, L. M., JUL 2008, "Self-Selection and Information Role of Online Product Reviews," Inf. Syst. Res, **19**(4), pp. 456–474.

[90] Rai, R., 2012, "Identifying Key Product Attributes and Their Importance Levels From Online Customer Reviews," International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers Digital Collection, Vol. 45028, pp. 533–540.

[91] Jin, J., Ji, P., and Gu, R., 2016, "Identifying Comparative Customer Requirements From Product Online Reviews for Competitor Analysis," Eng. Appl. Artif. Intell., **49**, pp. 61–73.