Topically diversified summarization of customer reviews

Florian Carichon and **Gilles Gaporossi**
HEC Montréal
3000, chemin de la Côte-Sainte-Catherine, Montréal (Québec), Canada, H3T 2A7.
florian.carichon@hec.ca
gilles.caporossi@hec.ca

Abstract

Promoting information coverage and sentence diversity is an efficient method to handle the fundamental issue of data heterogeneity or redundancy in multi-document summarization. We introduce a self-supervised algorithm for multi-document summarization that employs a multitask learning approach for topic diversification. Our model is based on two variational autoencoders that combine the training of a language model and a topic model to bias text generation and control the topic content of the produced summaries. We evaluate our method on the Amazon product review dataset and report ROUGE results and other metrics to assess information coverage. We demonstrate that our approach creates diversified outputs for the same batch of reviews and aspect-focused ones, allowing us to optimize text generation strategies.

1 Introduction

E-commerce and online sales platforms have grown substantially among the leading shopping media [1]. They change how we purchase products or services, allowing access to user experience. However, due to the subjective nature of reviews, customers must read many reviews to make an informed decision. By distilling the most important content in a reduced version of all opinions, automatic text summarization becomes crucial to help users.

The recent success of deep learning systems has led to significant improvement of extractive (Angelidis et al., 2021) or abstractive (See et al., 2017a; Paulus et al., 2017) document summarization models. With the domain-sensitive nature of product reviews, manufacturing large parallel corpora becomes costly and hardly transferable. Therefore, it has created a strong appetite for unsupervised summarization approaches where salient information depicts the consensual customer's point of view. However, the data heterogeneity of opinions distorts relevant content, resulting in overly broad summaries (Amplayo et al., 2021). Thus, It is essential to design strategies focusing on specific product aspects and transcribing this fine-grained content into the summary (Coavoux et al., 2019).

Since aspects can be implicitly grouped together according to themes, the detection of product review topics has naturally been associated with review aspects (Zhai et al., 2015). Methods such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and its deep learning variants have proven to be efficient in dynamically identifying these themes for opinion datasets (Ozyurt and Akcayol, 2021). In the context of opinion summarization, topic diversity increases the volume of the semantic space, improves information coverage, and therefore satisfies different needs of the user's population (Yogatama et al., 2015). The objective is then to optimize the sentences' topic relevancy and diversity (Li et al., 2010; Fang et al., 2015). Conditional variational autoencoders (Sohn et al., 2015) trained with topic modelling systems (Gao and Ren, 2019; Xiao et al., 2018) thus represent a promising avenue in this context.

In this article, we introduce an abstractive method for unsupervised customer opinion summarization that can produce text segments focused on various topics and combine them to maximize the input coverage. More specifically, our approach relies on a multi-task learning algorithm to train a topic and a language model jointly, both based on a variational autoencoder (VAE). We use the topic latent representation to condition the language model when learning review reconstruction. During the generation phase, we can select a subset of different topics to bias content included in the summary. We evaluated our approach on the Amazon product dataset, showing the importance of topic modelling to bring detailed and meaningful messages in such a heterogeneous context.

[1] https://www.forbes.com/advisor/business/ecommerce-statistics/

2 Related work

2.1 Multidocument Summarization for Opinion

Recent unsupervised abstractive techniques encapsulate information redundancy from a group of reviews into an average latent representation either directly (Chu and Liu, 2019; Bražinskas et al., 2020). However, such models suffer from aspects and topic heterogeneity, thus resulting in overly broad and almost irrelevant summaries. To address this issue, authors in (Angelidis and Lapata, 2018) create aspects-based representations with a partial autoencoder and devise an optimization function to select opinion that leverages their coverage. OpinionDigest (Suhara et al., 2020) is another method that clusters topically related reviews and employs a ranking algorithm to increase diversity in the output. Finally, (Amplayo et al., 2021) have introduced an interesting hybrid procedure that clusters opinions and extracts sentences to produce a summary predicated on popular or specific aspects. Regarding abstractive summarization, authors in (Coavoux et al., 2019) combine Meansum (Chu and Liu, 2019) with a clustering algorithm to conceive a latent representation for each group and form a text that maximizes input coverage. Our model is closely related because we modify the hierarchical VAE submitted in (Bražinskas et al., 2020) with a topic model. However, we propose a multi-task learning objective to produce dynamic topic representations, letting us condition the summary on the popular or specific topic/aspect.

2.2 Topic modeling

One of the most known and employed models is the Latent Dirichlet Allocation (LDA) (Blei et al., 2003) because of its generative ability and interpretability. The method has been applied in (Arora and Ravindran, 2008) for extractive summarization by submitting an algorithm that selects phrases with the highest probability of being produced by both the main topic and the document collection. Some authors have proposed increasing the coverage of the input texts by weighing the importance of the LDA topics with their similarity to ensure the diversity of sentences (Ren and de Rijke, 2015). Regarding recent deep learning models, some authors have adapted the re-parametrization trick of VAEs (Kingma and Welling, 2022) to multinomial distributions such as the Dirichlet distribution to create deep topic models (Srivastava and Sutton, 2017). Thereafter, such techniques have been used to obtain conditional language models to diversify sentence outputs. The idea is to produce biased latent representations by weighting input information by topics (Gao and Ren, 2019) or to concatenate directly the latent and the topic vectors (Xiao et al., 2018). Our approach combines these principles to learn relevant topics and optimize their selection for increasing opinion coverage in abstractive summarization.

3 Proposed Model

This section presents the general architecture of our multi-task learning approach as described in the figure 1. We first modify the hierarchical VAE summarizer proposed in (Bražinskas et al., 2020) by adding another VAE for topic modelling. We also introduce methods to select and condition summary generation regarding various topics.

The corpus is composed of customer reviews on different products. The vocabulary of the corpus is noted V. We define a batch of M customer reviews regarding a specific product as $\{R_1, ..., R_i, ..., R_M\}$ used to train our model. Each review R_i is composed of a set of words $X = \{X_1, ..., X_j, ..., X_N\}$, where N represents each review's variable length.

3.0.1 Topic Model

For a given review R_i, we apply a Bag of Words (BoW) encoding to obtain a vector BoW_i of size $|V|$, where dimensions indicate the word occurrence in R_i. This vector is then fed to a two-layer Forward Neural Network with a softplus activation function to create h_i^{bow}. We use this dense representation to encode the topic distribution through the continuous latent representation t_i. The objective of the model is to maximize the following:

$$\log \int \prod_{i=1}^{M} p_\theta(BoW_i|t_i, \beta) \tag{1}$$

where β represents the multinomial prior distribution of the topics over the vocabulary. As for the *ProdLDA* model (Srivastava and Sutton, 2017), we approximate the mixture of two multinomial distributions to their weighted multiplication. Therefore, we combine β and t_i to compute the probability of generating the output Bag of Words $BoW_i^{'}$:

$$p_\theta(BoW_i^{'}) = softmax([t_i \cdot \beta]) \tag{2}$$

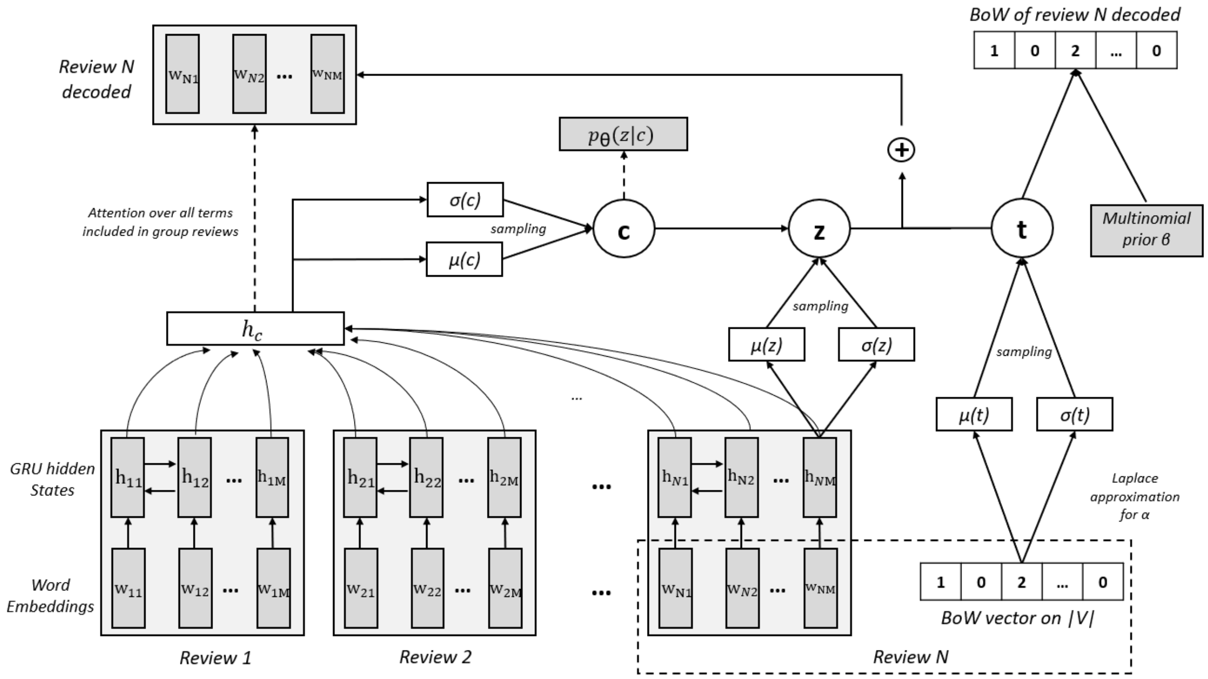

Figure 1: The multitask architecture for topic diversification of summary generation. The right part presents how the VAE is trained with a bag of word representation to obtain the topic distribution and the latent variable t. The left part displays the language model VAE. The latent variable c encodes the whole group of reviews while z encodes individual information. z is conditioned by c in training and is combined with t for the text reconstruction.

We train this part of the model with the mean square error function.

3.0.2 Language Model

We transform every input review with a pre-trained embedding model. The embedding matrix is fed to our encoder, a bidirectional Gated Recurrent Unit (GRU) (Cho et al., 2014). It produces an encoding h_{ij} for each word $j \in R_i$ and the last hidden state output h_{iN} used as the sentence representation. We based the training of our language model on the hierarchical VAE structure proposed in *Lsumm* (Bražinskas et al., 2020). Therefore, we first create the hidden representation h_c for all the group by computing the weighted sum over the attention of $m_{ij} = [h_{ij}; E_{ij}]$, the concatenation of the embedding and the GRU representations of the term x_j in R_i. We also assume a standard Gaussian distribution and apply a linear projection on h_c to sample the latent representation c encapsulating the information from the batch of reviews. Then, to perform the text reconstruction, we concatenate h_{iN}, the last GRU layer of R_i, and c to sample the latent variable z and pass it to our decoder.

When reconstructing, we perform N decoding steps to generate our sentence. We set the initial hidden state of the decoder, a simple GRU, s_0 to $[z_i; t_i]$ the concatenation of the topic and latent representation of R_i. At each decoding step t, we estimate the current hidden state s_t with the previous states s_{t-1} and predicted word $x^{'}_{t-1}$. We keep following the structure introduced in (Bražinskas et al., 2020) by calculating the attention distribution a^t, as in (Bahdanau et al., 2016), over the whole group of reviews R_i, excluding R_i. Once computed, we use every attention value $a^t_{\_i}$ to weight the representation $h_{\_i}$ of terms not belonging to R_i to create the context vector c_t. This vector is concatenated with the decoder state s_t and passed through a linear and a softmax layer to determine the probability of generating the output word $p_g(x^{'}_t)$:

$$P_g(x^{'}_t) = softmax(V^{'}(V[s_t, c^t] + b) + b^{'}) \quad (3)$$

where $V^{'}$, V, b, and $b^{'}$ are learnable parameters. We finally deploy a copy mechanism as presented in the *Pointer Generator Model* (PGN) (See et al., 2017b) to consider Out-Of-Vocabulary words. We compute the probability p_{gen} with a forward network and a sigmoid function over the context vector c_t, the hidden state s_t, and the previous predicted word $x^{'}_{t-1}$. The model uses p_{gen} to decide if it must preserve $x^{'}_{t}$ or to copy a term from R_i. The new probability then becomes:

$$P(x^{'}_{t}) = p_{gen} \times P_g(x^{'}_{t}) + (1 - p_{gen}) \times \sum_{i \in V_{ext}} (a^{t}_{\_i}) \tag{4}$$

where V_{ext} is the extended vocabulary aggregating the training vocabulary and the source document distribution. Finally, we let the model choose to draw terms directly from the distribution of topics $p(BoW^{'}_{i})$ defined in equation 3.0.1 by modifying the final probability:

$$P_{final}(x^{'}_{t}) = P(x^{'}_{t}) + p(BoW^{'}_{i}) \tag{5}$$

We empirically notice that letting the model choose between the two probabilities helps the model to converge better when learning the topic distribution. The language model is trained with the cross-entropy function.

3.0.3 General Architecture

Our complete approach combines the topic and the language models. The objective is to maximize the following function:

$$\log \int \left[p_\theta(c) \prod_{i=1}^{M} \int p_\theta(R_i | z_i, R\_i, BoW_i, t_i) \, p_\theta(z_i|c) dz_i \right] dc + \log \int \prod_{i=1}^{M} p_\theta(BoW_i | t_i, \beta) dt_i \tag{6}$$

The right part of the function describes the topic model, and the left part depicts our language model conditioned by the topic content. This approach enables the system to learn relevant topics and use them to condition summary generation.

3.1 Model distributions

This section describes the assumptions about the prior and posterior distributions. We rely on the principles defined in (Bražinskas et al., 2020) for approximating c and z and (Srivastava and Sutton, 2017) for t. We refer the lecturer to these articles for further mathematical details.

3.1.1 Reconstruction latent variables: c ans z

For c, we assume a standard normal prior distribution $p(c) = \mathcal{N}(c; 0, I)$. For the posterior distribution, we use the reparameterization trick (Kingma and Welling, 2022) for Gaussian distribution with a linear projection on h_c. We estimate the mean $\mu_\Phi(c)$ and variance $\sigma_\Phi(c)$ the approximated inference network $q_\Phi(c|h_c) = \mathcal{N}(c; \mu_\Phi(h_c), I\sigma_\Phi(h_c))$. Regarding z, we also assume a prior normal Gaussian distribution. The major difference is that the latter is conditioned by c to obtain $p_\theta(z|c) = N(z; \mu_\theta(c), I\sigma_\theta(c))$. As for the mean $\mu_\Phi(z)$ and the variance $\sigma_\Phi(z)$ of the inference posterior distribution, we use the same procedure by linearly projecting the concatenation $[R_i; c]$. Then, we sample z through $q_\Phi(z_i|R_i, c) = N(z_i; \mu_\Phi(R_i, c), I\sigma_\Phi(R_i, c))$.

3.1.2 Topic latent variable: t

We assume a Dirichlet prior distribution for the latent topic variable t because it has been shown beneficial to obtain good and interpretable topics (Blei et al., 2003). The reparameterization trick becomes a Laplace approximation with a softmax estimation to compute the distribution and make it tractable within the VAE framework. This approximation to the topic prior $p_\theta(t|\alpha)$ is equivalent to considering a logistic normal distribution with parameters with mean $\mu_\theta(t)$ and covariance matrix $\sigma_\theta(t)$ that are functions of α and K the number of defined topics. Once we assume this distribution, we can once again compute the parameters of the posterior distribution from an inference network as a linear projection on h^{BoW}_{i} to obtain $q_\Phi(t_i|h^{BoW}_{i}) = N(t_i; \mu_\Phi(h^{BoW}_{i}), I\sigma_\Phi(h^{BoW}_{i}))$.

3.2 Model loss function

We seek to maximize the Evidence Lower BOund (ELBO) for variational inference regarding the parameters θ and Φ. The following equations depict the language model noted $\mathcal{L}_{LM}$ and the topic model loss $\mathcal{L}_{TM}$.

$$\mathcal{L}_{LM}(\theta, \Phi) = \mathbb{E}_{q_\Phi(c|R)} \left[\sum_{i=1}^{M} \mathbb{E}_{q_\Phi(z_i|R_i, c)} [\log p_\theta(R_i | z_i, t_i, BoW_i)] - \sum_{i}^{M} \mathbb{D}_{KL} \left[q_\Phi(z_i|R_i, c) || p_\theta(z_i|c) \right] \right] - \mathbb{D}_{KL} \left[q_\Phi(c|R) || p_\theta(c) \right] \tag{7}$$

$$\mathcal{L}_{TM}(\theta,\Phi) = \sum_{i=1}^{M} \mathbb{E}_{q_\Phi(t_i|BoW_i)} [\log p_\theta(BoW_i|t_i,\beta) \\ -\mathbb{D}_{KL}[q_\Phi(t_i|BoW_i)||p_\theta(t_i|\alpha)]] \quad (8)$$

For both losses, the left part of the expressions ensures the text reconstruction of R_i or its bag of words representation BoW_i. The right term is the *Kullback-Leibler* divergence, which guarantees to match our prior distributions. We then minimize the joint loss as the sum of $\mathcal{L}_{LM}$ and $\mathcal{L}_{TM}$.

3.3 Summary Generation

To condition summary generation, we must first set up a strategy to designate the $k = [1, ..., K]$ main theme(s) on which to focus. We determine the relevant topics by identifying the ones that deviate the most from their expected prior distribution (AlSumait et al., 2009). To ensure further their diversity, we have implemented a Maximum Margin Relevance approach (Carbonell and Goldstein, 1998). Therefore, we choose topics from the posterior distribution that maximize $cos(t_k^{prior}, t_k) - \lambda * cos(t_k, t_j)$, where t_k is the topic distribution over our documents, t_j are the already picked ones, cos is the cosine similarity, and $\lambda = 0.5$.

For each selected topic k, we bias the hidden representation h_c with the posterior topic-word distribution β_k. We establish the set $X_{topics}k$ by preserving $1/8$ of the most topically probable terms in β_k from the extended vocabulary. We tested multiple filtering factors ranging from $1/2$ to $1/32$. Our first observations let us think that if we keep too many words, we do not impose enough diversity in the outputs, and if we remove too much, sentences become ungrammatical. Therefore, we empirically chose to preserve $1/8$ words as a good balance between the produced summaries' diversity and coherency. When creating h_c, instead of attending to all the group reviews' words, we attend only to $X_{topics}k$; the remaining words are masked. Then, we fix c to $\mu_\Phi(c)$ constructed via the inference model through this biased h_c.

To further condition the summarization of our text collection, we use the topic distribution t_k to set z to $z^{topic} = \mu_\theta(z) * t_k$, a topically biased representation of its prior mean for each document. We sample our summary by maximizing the probability expectation $P(x_t^{'})$ only. We instead apply $p(BoW_i^{'})$ in the beam search method to select among our K best-generated hypotheses the one that maximizes the sum of the two probabilities.

4 Experiment

4.1 Dataset

We trained our model on the Amazon Product dataset composed of reviews on 29 product categories (He and McAuley, 2016). We have considered products with at least 15 and a maximum of 100 reviews. We excluded texts under 8 and above 200 tokens. We remove the ones above the 90^{th} percentile each time. Since we aim to demonstrate the model's ability to handle heterogeneous information, we sample reviews from 19 categories and evaluate the model on the same 200 human-generated summaries as in (Bražinskas et al., 2020). Our final training data is composed of 17,497 reviews drawn from 303 products and the validation of 3,105 reviews from 50 products.

4.2 Implementation details

Our model uses the GloVe 200-dimensional pre-trained word embeddings (Pennington et al., 2014). The text was lowercased, and we used Spacy tokenizer and part-of-speech tagger [2] to preserve only adverbs, adjectives and nouns for the BOW representation. Both the model's encoder and decoder are composed of a single bidirectional layer with a size of 512 hidden units. We set the dimensions of the latent variable z and c to 600. We set the number of topics t to 30. We initialize the model during training with a Xavier uniform distribution (Glorot and Bengio, 2010). We trained the model for 150 epochs with the Adam optimizer (Kingma and Ba, 2017), a learning rate of $5 * 10^{-4}$, a weight decay of 10^{-6}, a gradient clipping of 10, and a dropout ration of 0.2. Regarding the KL divergence terms, we have employed a cycling function with $r = 0.8$ (Fu et al., 2019) and a maximum value of 1 for z and 0.65 for c. We have used a linear scheduling function between epochs 0 to 40 with a max value set to 1 for t. Finally, we apply the beam search method with a beam size established to 5 and an n-gram blocking method (Paulus et al., 2017) set to avoid trigram repetitions. Our code is available

[2] https://spacy.io/

on GitHub [3].

4.3 Evaluation

We compare our approach with 3 different baselines. The first is *BERT for Text Summarization* (Miller, 2019), and the second is *TextRank* (Mihalcea and Tarau, 2004). These two extractive methods are regularly used as baselines for evaluating general-purpose summarization. We also compare to our unsupervised base abstractive model *Lsumm* (Bražinskas et al., 2020). We trained and fine-tuned it on our Amazon dataset with the same parameters detailed in the section 4.2.

We report the average and maximum ROUGE F1 scores (Lin, 2004) for the different baselines on the evaluation dataset, which encompasses 3 human-created summaries for 60 products consisting of 8 reviews. We also provide the ROUGE scores with filtered stop words to emphasize the presence of content words in the generated outputs. We further include BLEURT scores (Sellam et al., 2020) to indicate to what extent the summaries convey the meaning of the input. Finally, we disclose how well methods can capture the topics addressed in the opinions expressed. To that extent, we train a LDA model with the Gensim library [4] on our training dataset. We then measure the similarity of the topic distributions and the semantic coherence of topics as described in (Greene et al., 2014) between the input reviews and the produced summaries.

5 Results and analysis

5.1 Model evaluation

We introduce two models based on our approach. The first method, *TopiCatSumm*, generates one summary based on K topically conditioned sentences of length N_{mean}/K, where N_{mean} is the average length of a batch of reviews. Since $N_{mean} = 58$ words, we set $K = 3$ to ensure diversity while generating long enough texts to be coherent. For the second, *TopicNSumm*, we have duplicated the evaluation dataset by making 3 topically distinct outputs of length N_{mean}. We report in the table 1 both the average score between the summary and all the references and the maximum score with its best matching reference. In the case of our second configuration *TopicNSumm*, we first pair each human production with the summary that optimizes its ROUGE score, and we report the average and maximum for all associated metrics.

Contrary to previous observations, self-supervised abstractive approaches appear worse than unsupervised extractive ones. This result likely represents the issues created by increasing data heterogeneity in the training set. The results also show that *TopicNSumm* allows an efficient optimization for matching related summaries with their reference. *TopiCatSumm* was the least performing, partly due to the size constraint penalizing the production of coherent sequences, but both methods improve the topic diversity and content coverage. We exhibit further these observations in the table 2.

These results reveal that both our approaches significantly improve content coverage and the topic distribution of the original customer opinions compared to the base abstractive approach. The filtered ROUGE scores further emphasize that our methods improve the ability to generate meaningful material. We assume that the performance of extractive strategies remains high because of the intrinsic homogeneity of a batch dealing with the same product. Therefore, we conduct a quick analysis of ROUGE for a batch of 16 reviews from 2 distinct products. Once again, we pair the best matching results to the summary to disclose average and maximum scores in the table 3. Our approaches suffer less from increasing heterogeneity, especially compared to extractive approaches, where the drop is the most important. In future studies, We plan to evaluate our model's capability to handle these extreme cases.

Finally, we provide some examples of generated documents by our model and the various baselines in appendix Appendix A..

5.2 Model and configuration analysis

We integrated our topic model with our language and summarization model during the training stage. We used a BOW vector for each review in our approach, but we could have created one for the entire group instead. However, by doing so, we observe that the model is unable to optimize both $\mathcal{L}_{TM}$ and $\mathcal{L}_{LM}$ at the same time. The need to capture individual and group information either restricts too much or brings too much noise into the latent variable t, penalizing the language or the topic model. During training, we also directly

[3] https://github.com/fcarichon/TopicDiversifiedVAESumm

[4] https://radimrehurek.com/gensim/models/ldamodel.html

Table 1: ROUGE scores on the Amazon dataset

Methods	R-1 (avg)	R-1 (max)	R-2 (avg)	R-2 (max)	R-L (avg)	R-L (max)
BERT Summarizer	25.03	30.33	4.17	7.39	15.31	18.67
TextRank	29.42	34.87	5.1	8.36	16.82	20.17
LSumm	17.57	21.92	0.51	1.14	10.91	13.59
TopiCatSumm	16.91	20.32	0.34	8.83	9.75	11.79
TopicNSumm	19.64	23.24	0.78	1.9	11.58	13.9

Table 2: Topic content and coverage evaluation on Amazon dataset

Methods	R-1 filt. (avg)	R-1 filt. (max)	BLEURT	Word topic overlap	topic similarity
Human references	NA	NA	-0.464	0.95	0.488
BERT Summarizer	18.64	25.04	-0.774	0.469	0.336
TextRank	21.24	27.29	-0.673	0.521	0.338
LSumm	6.67	9.89	-0.889	0.201	0.2
TopiCatSumm	9.39	12.71	-0,579	0.494	0.383
TopicNSumm	11.58	15.53	-0,677	0.678	0.477

concatenate z and t, but it would be tempting to do it for c and t since we use this representation as a condition for the summary generation. However, it results again in a significant drop for both losses, thus in low topic quality and inability to create diverse outputs. Including t in early layers makes it possible for the model to bypass learning the topic distribution, and it does not facilitate the task of capturing information for z as observed in (Xiao et al., 2018).

We have tested several configurations to bias the summary topically. The first experiments and the study of the output texts have emphasized the importance of employing the posterior distribution for sampling t and β matrices. With the data heterogeneity, using prior distributions leads to inducing broad topics, thus decreasing content quality and increasing hallucinations. The remaining iterative tests modify parameters in our current setup, stated as configuration 0 hereafter and described in section 3.3.

- Configuration 1: As for c and t, we have set t at its mean $mu(t)$ only.
- Configuration 2: We have tried to bias c with the main topic distribution by creating $c^{topic} = mu_\Phi(c) * t_k$ as we do for z^{topic}.
- Configuration 3: Inversely, instead of employing a topic biased z^{topic}, we have set it to its mean $z = mu_\theta(z)$ as in *Lsumm*.
- Configuration 4: Rather than masking the attention for h_c, we could weight the attention tensor with the word's topic probability.
- Configuration 5.a: Rather than masking attention at the group level, we have masked attention used directly in the decoder.
- Configuration 5.b: As for configuration 4, we have also tried to weight the decoder's attention tensor with the word's topic probability rather than masking it.
- Configuration 6: We have employed the BOW probability $p(BoW_i^{'})$ for the summary generation.

We report the ROUGE-1 and BLEURT results for the *TopicNSumm* model. We also provide a diversity metric to emphasize issues met by some configurations. To that end, we re-encode the generated summaries, and then measure the average cosine distance between these encodings. The table 4 displays the results obtained.

Results from configuration 1 emphasize again the value of having a precise and rich topic distribution to draw effectively relevant information from the topic distribution. The absence of difference in configuration 2 and the significant decrease of summaries' diversity in configuration 3 confirms the importance of biasing z as in training and not the group representation c, where the language model might compensate for the topic conditioning. The BLEURT and diversity scores of configuration 4 corroborate this hypothesis since implementing a soft bias, such as weighting the attention, is not enough to produce heterogeneous outputs. We can also note from analysis of configurations 5.a, 5.b, and 6 that directly impacting the text generation with topic distribution, in the decoder or the final probability distribution, is effective for producing relevant content. However, it comes at the expense of the summary coherency and readability.

Finally, another possibility is to let users bias the summary toward specific topics by defining

Table 3: Evaluation of the various approaches summarizing batches of 16 reviews sampled from 2 different products categories of the Amazon dataset.

Methods	R-1 (avg)	R-1 (max)	R-1 filt. (avg)	R-1 filt. (max)
BERT Summarizer	18.63	25.04	13.35	21.96
TextRank	21.24	27.29	14.02	23.58
LSumm	18.39	25.58	4.13	6.17
TopiCatSumm	16.67	22.17	9.11	15.17
TopicNSumm	18.45	25.15	11.04	18.02

Table 4: Table introducing the different results from various model configurations. We repeat the results of our main model in the first line for comparison.

TopicNSumm configurations	R-1 (avg)	R-1 (max)	BLEURT	Hidden diversity
Configuration 0	19.64	23.24	-0.677	0.578
Configuration 1	16.76	20.23	-0.656	0.513
Configuration 2	19.62	22.58	-0.69	0.534
Configuration 3	19.58	23.12	-0.65	0.328
Configuration 4	19.53	23.56	-0.72	0.469
Configuration 5.a	19.82	23.86	-0.63	0.557
Configuration 5.b	19.78	23.92	-0.61	0.558
Configuration 6	19.78	23.39	-0.677	0.562

their set of keywords $X^{user} = X_0^{user}, ..., X_U^{user}$. In that case, we identify the U main topics that maximize the probability $p(X^{user}|t_u)$ in the topic-word matrix. We provide 3 examples in table 7 in appendix Appendix B. of summaries generated by inputting the term “price” in the appendix. We observe that the model has conditioned the texts to include terms such as “expensive”, “full cost”, or even “budget”, which relate to the price. We also note that the model cannot bias the summary if the reviews do not deal with the input term. While this can be frustrating for the user, it is beneficial that the model does not hallucinate false information.

5.3 Limitations and future research avenues

The first limitation of our approach comes from the additional hyperparameters we introduced. We had to fine-tune many variables and distributions to make the model efficient. Specifically, we noticed that the number of topics selected is crucial since it influences the output quality and is, unfortunately, domain- or product-dependent. The second impediment of our method can be generalized to every system that tries to bias text generation. Indeed, biasing language models can lead to predicting terms that should not have been otherwise, inducing a potential loss of coherence or unwanted hallucinations. Finally, we are aware of the limitations of our architecture based on single-layer RNNs. The text coherency is inferior to current models predicated on pre-trained large language models (LLMs). Beyond the problems of budget and access to sufficiently powerful machines, studying simpler models guarantees that the capacity of these architectures does not absorb our approach and does induce diversity. We leave the analysis of its application to LLMs for future work.

6 Conclusion

In this paper, we introduced an unsupervised topic method for multi-document summarization of product reviews. It relies on two variational autoencoders combined in a multitask learning objective. This approach improves abstractive summarization models' performance by increasing content coverage or focusing on specific important topics. With this research, we hope that we have successfully demonstrated that this model could enhance the capacity of generative large language models to handle heterogeneous data and bias and diversify their outputs.

References

Loulwah AlSumait, Daniel Barbará, James Gentle, and Carlotta Domeniconi. 2009. Topic significance ranking of lda generative models. In *Machine Learning and Knowledge Discovery in Databases*, pages 67–82, Berlin, Heidelberg. Springer Berlin Heidelberg.

Reinald Kim Amplayo, Stefanos Angelidis, and Mirella Lapata. 2021. Aspect-controllable opinion summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6578–6593, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Stefanos Angelidis, Reinald Kim Amplayo, Yoshihiko Suhara, Xiaolan Wang, and Mirella Lapata. 2021. Extractive opinion summarization in quantized transformer spaces. *Transactions of the Association for Computational Linguistics*, 9:277–293.

Stefanos Angelidis and Mirella Lapata. 2018. Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3675–3686, Brussels, Belgium. Association for Computational Linguistics.

Rachit Arora and Balaraman Ravindran. 2008. Latent dirichlet allocation based multi-document summarization. AND '08, page 91–97, New York, NY, USA. Association for Computing Machinery.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural machine translation by jointly learning to align and translate.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3(null):993–1022.

Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2020. Unsupervised opinion summarization as copycat-review generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5151–5169, Online. Association for Computational Linguistics.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 335–336, New York, NY, USA. Association for Computing Machinery.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Eric Chu and Peter Liu. 2019. MeanSum: A neural model for unsupervised multi-document abstractive summarization. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1223–1232. PMLR.

Maximin Coavoux, Hady Elsahar, and Matthias Gallé. 2019. Unsupervised aspect-based multi-document abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 42–47, Hong Kong, China. Association for Computational Linguistics.

Hanyin Fang, Weiming Lu, Fei Wu, Yin Zhang, Xindi Shang, Jian Shao, and Yueting Zhuang. 2015. Topic aspect-oriented summarization via group selection. *Neurocomputing*, 149:1613–1619.

Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. 2019. Cyclical annealing schedule: A simple approach to mitigating KL vanishing. 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 240–250, Minneapolis, Minnesota. Association for Computational Linguistics.

Ce Gao and Jiangtao Ren. 2019. A topic-driven language model for learning to generate diverse sentences. *Neurocomputing*, 333:374–380.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256, Chia Laguna Resort, Sardinia, Italy. PMLR.

Derek Greene, Derek O'Callaghan, and Pádraig Cunningham. 2014. How many topics? stability analysis for topic models. In *Machine Learning and Knowledge Discovery in Databases*, pages 498–513, Berlin, Heidelberg. Springer Berlin Heidelberg.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 507–517, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization.

Diederik P Kingma and Max Welling. 2022. Auto-encoding variational bayes.

Xuan Li, Yi-Dong Shen, Liang Du, and Chen-Yan Xiong. 2010. Exploiting novelty, coverage and balance for topic-focused multi-document summarization. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, page 1765–1768, New York, NY, USA. Association for Computing Machinery.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain. Association for Computational Linguistics.

Derek Miller. 2019. Leveraging bert for extractive text summarization on lectures.

Baris Ozyurt and M. Ali Akcayol. 2021. A new topic modeling based approach for aspect extraction in aspect based sentiment analysis: Ss-lda. *Expert Systems with Applications*, 168:114231.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Zhaochun Ren and Maarten de Rijke. 2015. Summarizing contrastive themes via hierarchical nonparametric processes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, page 93–102, New York, NY, USA. Association for Computing Machinery.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017a. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017b. Get to the point: Summarization with pointer-generator networks.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models.

Yoshihiko Suhara, Xiaolan Wang, Stefanos Angelidis, and Wang-Chiew Tan. 2020. OpinionDigest: A simple framework for opinion summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5789–5798, Online. Association for Computational Linguistics.

Yijun Xiao, Tiancheng Zhao, and William Yang Wang. 2018. Dirichlet variational autoencoder for text modeling.

Dani Yogatama, Fei Liu, and Noah A. Smith. 2015. Extractive summarization by maximizing semantic volume. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1961–1966, Lisbon, Portugal. Association for Computational Linguistics.

ChengXiang Zhai, William W. Cohen, and John Lafferty. 2015. Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. *SIGIR Forum*, 49(1):2–9.

Appendix A. Models' generated texts

The table 5 presents the produced results by the different models for batches of 8 reviews. We note the better coherence and quality of the extractive baseline. However, we can also observe for vacuum filter examples that our method generated texts on the eating system, the filters and their price, or the fans. It highlights the ability of our model to increase the coverage of the inputs' topics and aspects. The table 6 shows the generated texts for a batch of 16 documents. The benefit of our approach is even more obvious here when we see 2 summaries focusing on the vacuum and the other on the steamer. In contrast, our baselines cannot manage this information diversity and have a considerable loss of coherency and relevance.

Appendix B. Texts with an input keyword

The table 7 presents the produced summaries by our model when we provide an input term to bias the generation of the model. A summary is then generated for each given term. The products presented here are the ones used in the previous examples to allow output comparison with this new bias.

Table 5: Table with examples of generated texts. For each product we provide the text generated by our two configurations, the absractive model LSumm and the extractive model TextRank.

B0002U34HY (CHV1510 Vacuum filter)	Our model TopiCatSumm	Easy fix before expected not much monster filters but with regular use handles clean, seems sturdy. However this filter was difficult with product support, I read comparable CHV1510 on here as other. The dirty class hitting washable model construction of functionality CHV1510 ridiculous, quality functionality washable.
	Our model TopicNSumm summary 1	CHV1510 games was home from eating all i complained without such cool 3rd CHV1510 brand I and amount on them off position not one time with filter that are just guessing all color!
	Our model TopicNSumm summary 2	that said filter and cheaper on shipping as hair fast shipping here than what should is but for something changed after working. The filter holder showed that, what appears it properly had different place for filter like using generic brand at all!
	Our model TopicNSumm summary 3	For the fans mounted cold lights: positive copies filters the world has broken open when aid properly from CHV1510, so in some amounts source on wrench breaking during these are fantastic and I still recommend
	LSumm	it says harder. to install with filter as possible for filter! it takes some amounts. it seems too strong as opposed the original one of it and
	TextRank	This is the wrong filter if you are buying the CHV1510 Hand Vacuum. This item list listed with the vacuum – 'frequently bought together' with the Black & Decker CHV9608 9.6 Volt Cyclonic-Action Cordless DustBuster BUT this filter does NOT fit!

Table 5: Table with examples of generated texts (Continued)

B0013EQ20Y (Frye Boots)	Our model TopiCatSumm	it wish my face soft hat, the boots it cozy lifts up nice. Comfy ugg Frye perfectly residue inside comfortable stretchy amounted just what the doctor ordered from boots all. I served comfy, boot though sticks right but quickly to safety snug evenly over, all socks together is
	Our model TopicNSumm summary 1	Indeed an excellent product and most excellent boots base and nice as wide in between all sizes up. It needs enough for all occasion beware of adjustments such all over cameras during!
	Our model TopicNSumm summary 2	Indeed comfy! Securely packaged, the it too and I am wearing! it makes great for heavy use thick rooms but tough construction and comfort, sound nicely tasted Frye but
	Our model TopicNSumm summary 3	comfy boots has already hanging down set I wish where had them on fire if there have many on bugs like paper itself while having. Overall this pair work well
	LSumm	it seems so sturdy enough like that is. it seems more sturdy than expected to get them again and was worth to try them! it seems more comfortable! it seems better with
	TextRank	they can be a beast to get on, like any boot fit to last; once on, they are incredibly comfortable. With a 20year break from not wearing Frye it was a pleasant surprise the quality has stood the test of time.

Table 6: Table with examples of generated text by the various models for batches of 16 reviews sampled from 2 different products of two different categories.

B0002U34HY vacuum filter & B00006IUVM kitchen steamer	Our model TopicNSumm summary 1	quality filters not do any reviews and picture looks as usual but for decades material seems fine but great purchase and deliver quality packaged! yeah and trust with
	Our model TopicNSumm summary 2	quality filter for many light steamer washable rice brand steamer, although is just easy enough without sending to play using without issues until much sized goes steamer easy too steam rice for each nut only goes straight smoothly
	Our model TopicNSumm summary 3	ladies! steam it has superior points of shelves from there : do something that? this steamer gives all aspects go, some kind opened without wearing them into this. So in some reviews from dragon appeared steam as directed, received mine ripped rice vegetables today
	LSumm	the filter is just what i needed. i have a lot of the filter and the filter. is not the same as the original filter.. is a great deal. is a great deal. is a great deal. is a very a very a very
	TextRank	This is the wrong filter if you are buying the CHV1510 Hand Vacuum. Sometimes I use the steamer for just one vegetable, or for rice, but it’s really nice to have the separate basket.

Table 7: Examples of generated texts by our TopicNSumm where we input the word "price" to the model.

B0013EQ20Y (Frye Boots)	comfy noticeable! easy boots comfortable leather is inexpensive and wonderfully easy quality although is heavy as long to high although! instead i do wish that i have ordering it or worn on amazon.com since that it broke in two, only bought it 4 and times full cost
B00006IUVM (Kitchen Steamer)	updated hard 3 days! steam as use to force me rice is perfect with all customers at work budget is able with hesitant help at night supply store, too expensive than to sell items.
B0002U34HY (CHV1510 Vacuum filter)	CHV1510 filters is too and save dust the legs on top because occasionally leave volume under cycle i make sure look for washable filter or something. maybe it only keeps wet VF08